MT7623N Datasheet for Development Board

Version: 1.1
Release date: 2019-9-18

Document Revision History

Revision	Date	Author	Description
0.1	2017-03-27	Jia Chen	First released
1.0	2017-04-07	Leon Chung	Update ETHSYS, HIFSYS
1.1	2019-09-18	Steven Liu	Update ETHSYS

Table of Contents

Contents

List of Figures

List of Tables

1 GPIO

1.1 Introduction

By setting up the control registers, the MCU software can control the direction, output value and read the input values on these pins. The GPIOs are multiplexed with other functions to reduce the pin count.

1.2 Block Diagram

Below figure is the block diagram of GPIO. Each GPIO controls the auxlillary mode by programming GPIO_MODE_SELx command register. Besides, the dedicated register bits can be set to 1 or 0 by writing the bits of GPIO_MODE_SETx or GPIO_MODE_RESETx to 1.
GPIO_DIR, GPIO_DOUT and GPIO_PULLEN are also programmable by the same method of GPIO_MODE.

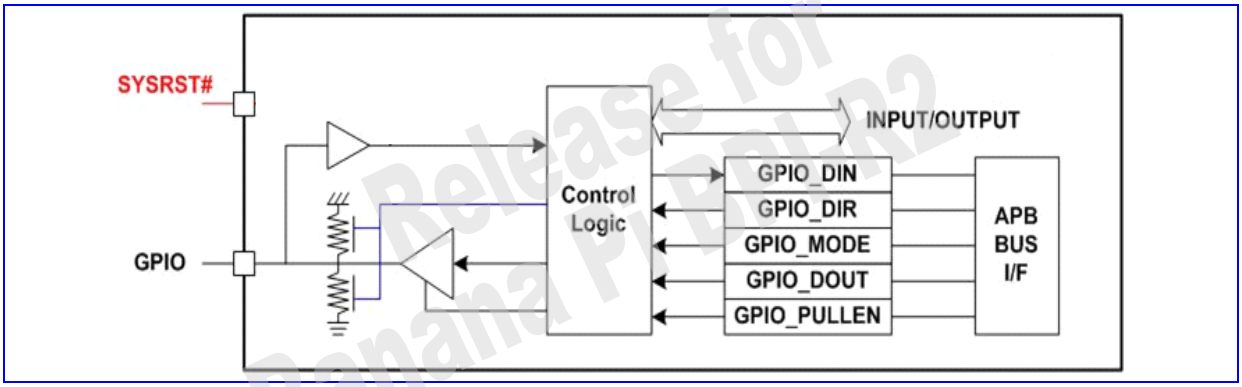


Figure 1-1: Block Diagram of GPIO

1.3 Aux Funtion

Below table is the aux. function of each GPIO, showing the mapping table of aux.name, mode number, CU/CD(controllable pull-up and pull-down) and driving capability.

Table 1-1: Aux. Function Table of GPIO

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
PWRAP_SPI0_MI	0	GPIO0	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_SPIDO	IO	CU,CD	2/4/6/8mA	-
	2	PWRAP_SPIDI	IO	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_SPI0_MO	0	GPIO1	IO	CU,CD	2/4/6/8mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	1	PWRAP_SPIDI	IO	CU,CD	2/4/6/8mA	-
	2	PWRAP_SPIDO	IO	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_INT	0	GPIO2	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_INT	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_SPI0_CK	0	GPIO3	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_SPICK_I	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_SPI0_CSN	0	GPIO4	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_SPICS_B_I	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_SPI0_CK2	0	GPIO5	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_SPICK2_I	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	ANT_SEL1	O	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
PWRAP_SPI0_CSN2	0	GPIO6	IO	CU,CD	2/4/6/8mA	-
	1	PWRAP_SPICS2_B_I	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	ANT_SEL0	O	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[0]	IO	CU,CD	2/4/6/8mA	-
SPI1_CSN	0	GPIO7	IO	CU,CD	2/4/6/8mA	-
	1	SPI1_CS	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	4	KCOL0	IO	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[12]	IO	CU,CD	2/4/6/8mA	-
SPI1_MI	0	GPIO8	IO	CU,CD	2/4/6/8mA	-
	1	SPI1_MI	I	CU,CD	2/4/6/8mA	-
	2	SPI1_MO	O	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	KCOL1	IO	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[13]	IO	CU,CD	2/4/6/8mA	-
SPI1_MO	0	GPIO9	IO	CU,CD	2/4/6/8mA	-
	1	SPI1_MO	O	CU,CD	2/4/6/8mA	-
	2	SPI1_MI	I	CU,CD	2/4/6/8mA	-
	3	EXT_FRAME_SYNC	I	CU,CD	2/4/6/8mA	-
	4	KCOL2	IO	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[14]	IO	CU,CD	2/4/6/8mA	-
RTC32K_CK	0	GPIO10	IO	CU,CD	2/4/6/8mA	-
	1	RTC32K_CK	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
WATCHDOG	0	GPIO11	IO	CU,CD	2/4/6/8mA	-
	1	WATCHDOG	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
SRCLKENA	0	GPIO12	IO	CU,CD	2/4/6/8mA	-
	1	SRCLKENA	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
SRCLKENAI	0	GPIO13	IO	CU,CD	2/4/6/8mA	-
	1	SRCLKENAI	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	7	-	-	CU,CD	2/4/6/8mA	-
URXD2	0	GPIO14	IO	CU,CD	4/8/12/16mA	-
	1	URXD2	I	CU,CD	4/8/12/16mA	-
	2	UTXD2	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	SRCCLKENAI2	I	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[30]	IO	CU,CD	4/8/12/16mA	-
UTXD2	0	GPIO15	IO	CU,CD	4/8/12/16mA	-
	1	UTXD2	O	CU,CD	4/8/12/16mA	-
	2	URXD2	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[31]	IO	CU,CD	4/8/12/16mA	-
PCM_CLK	0	GPIO18	IO	CU,CD	4/8/12/16mA	-
	1	PCM_CLK0	IO	CU,CD	4/8/12/16mA	-
	2	MRG_CLK	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	MM_TEST_CK	I	CU,CD	4/8/12/16mA	-
	5	CONN_DSP_JCK	I	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_CLKO	IO	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[3]	IO	CU,CD	4/8/12/16mA	-
PCM_SYNC	0	GPIO19	IO	CU,CD	4/8/12/16mA	-
	1	PCM_SYNC	IO	CU,CD	4/8/12/16mA	-
	2	MRG_SYNC	IO	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	CONN_DSP_JINTP	O	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_SYNC	IO	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[5]	IO	CU,CD	4/8/12/16mA	-
PCM_RX	0	GPIO20	IO	CU,CD	4/8/12/16mA	-
	1	PCM_RX	I	CU,CD	4/8/12/16mA	-
	2	MRG_RX	I	CU,CD	4/8/12/16mA	-
	3	MRG_TX	O	CU,CD	4/8/12/16mA	-
	4	PCM_TX	O	CU,CD	4/8/12/16mA	-
	5	CONN_DSP_JDI	I	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_RX	I	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[4]	IO	CU,CD	4/8/12/16mA	-
PCM_TX	0	GPIO21	IO	CU,CD	4/8/12/16mA	-
	1	PCM_TX	O	CU,CD	4/8/12/16mA	-
	2	MRG_TX	O	CU,CD	4/8/12/16mA	-
	3	MRG_RX	I	CU,CD	4/8/12/16mA	-
	4	PCM_RX	I	CU,CD	4/8/12/16mA	-
	5	CONN_DSP_JMS	I	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_TX	O	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[2]	IO	CU,CD	4/8/12/16mA	-
EINT0	0	GPIO22	IO	CU,CD	4/8/12/16mA	-
	1	UCTS0	I	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	2	-	-	CU,CD	4/8/12/16mA	-
	3	KCOL3	IO	CU,CD	4/8/12/16mA	-
	4	CONN_DSP_JDO	O	CU,CD	4/8/12/16mA	-
	5	EXT_FRAME_SYNC	I	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[30]	IO	CU,CD	4/8/12/16mA	-
EINT1	0	GPIO23	IO	CU,CD	4/8/12/16mA	-
	1	URTS0	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	KCOL2	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_TDO	O	CU,CD	4/8/12/16mA	-
	5	EXT_FRAME_SYNC	I	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[29]	IO	CU,CD	4/8/12/16mA	-
EINT2	0	GPIO24	IO	CU,CD	4/8/12/16mA	-
	1	UCTS1	I	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	KCOL1	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_DBGACK_N	O	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[28]	IO	CU,CD	4/8/12/16mA	-
EINT3	0	GPIO25	IO	CU,CD	4/8/12/16mA	-
	1	URTS1	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	KCOL0	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_DBGI_N	I	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[27]	IO	CU,CD	4/8/12/16mA	-
EINT4	0	GPIO26	IO	CU,CD	4/8/12/16mA	-
	1	UCTS3	I	CU,CD	4/8/12/16mA	-
	2	DRV_VBUS_P1	O	CU,CD	4/8/12/16mA	-
	3	KROW3	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_TCK0	I	CU,CD	4/8/12/16mA	-
	5	CONN_MCU_AICE_JCKC	I	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[26]	IO	CU,CD	4/8/12/16mA	-
EINT5	0	GPIO27	IO	CU,CD	4/8/12/16mA	-
	1	URTS3	O	CU,CD	4/8/12/16mA	-
	2	IDDIG_P1	I	CU,CD	4/8/12/16mA	-
	3	KROW2	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_TDI	I	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[25]	IO	CU,CD	4/8/12/16mA	-
EINT6	0	GPIO28	IO	CU,CD	4/8/12/16mA	-
	1	DRV_VBUS	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	KROW1	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_TRST_B	I	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[24]	IO	CU,CD	4/8/12/16mA	-
EINT7	0	GPIO29	IO	CU,CD	4/8/12/16mA	-
	1	IDDIG	I	CU,CD	4/8/12/16mA	-
	2	MSDC1_WP	I	CU,CD	4/8/12/16mA	-
	3	KROW0	IO	CU,CD	4/8/12/16mA	-
	4	CONN_MCU_TMS	I	CU,CD	4/8/12/16mA	-
	5	CONN_MCU_AICE_JMSC	IO	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[23]	IO	CU,CD	4/8/12/16mA	-
I2S1_DATA	0	GPIO33	IO	CU,CD	4/8/12/16mA	-
	1	I2S1_DATA	IO	CU,CD	4/8/12/16mA	-
	2	I2S1_DATA_BYPS	O	CU,CD	4/8/12/16mA	-
	3	PCM_TX	O	CU,CD	4/8/12/16mA	-
	4	IMG_TEST_CK	I	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_TX	O	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[8]	IO	CU,CD	4/8/12/16mA	-
I2S1_DATA_IN	0	GPIO34	IO	CU,CD	4/8/12/16mA	-
	1	I2S1_DATA_IN	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_RX	I	CU,CD	4/8/12/16mA	-
	4	VDEC_TEST_CK	I	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_RX	I	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[7]	IO	CU,CD	4/8/12/16mA	-
I2S1_BCK	0	GPIO35	IO	CU,CD	4/8/12/16mA	-
	1	I2S1_BCK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_CLK0	IO	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_CLKO	IO	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[9]	IO	CU,CD	4/8/12/16mA	-
I2S1_LRCK	0	GPIO36	IO	CU,CD	4/8/12/16mA	-
	1	I2S1_LRCK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_SYNC	IO	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_PCM_SYNC	IO	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[10]	IO	CU,CD	4/8/12/16mA	-
I2S1_MCLK	0	GPIO37	IO	CU,CD	4/8/12/16mA	-
	1	I2S1_MCLK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[11]	IO	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
JTMS	0	GPIO39	IO	CU,CD	2/4/6/8mA	-
	1	JTMS	IO	CU,CD	2/4/6/8mA	-
	2	CONN_MCU_TMS	I	CU,CD	2/4/6/8mA	-
	3	CONN_MCU_AICE_JMSC	IO	CU,CD	2/4/6/8mA	-
	4	DFD_TMS_XI	I	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
JTCK	0	GPIO40	IO	CU,CD	2/4/6/8mA	-
	1	JTCK	I	CU,CD	2/4/6/8mA	-
	2	CONN_MCU_TCK1	I	CU,CD	2/4/6/8mA	-
	3	CONN_MCU_AICE_JCKC	I	CU,CD	2/4/6/8mA	-
	4	DFD_TCK_XI	I	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
JTDI	0	GPIO41	IO	CU,CD	2/4/6/8mA	-
	1	JTDI	I	CU,CD	2/4/6/8mA	-
	2	CONN_MCU_TDI	I	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	DFD_TDI_XI	I	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
JTDO	0	GPIO42	IO	CU,CD	2/4/6/8mA	-
	1	JTDO	O	CU,CD	2/4/6/8mA	-
	2	CONN_MCU_TDO	O	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	DFD_TDO	O	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-
NCLE	0	GPIO43	IO	CU,CD	4/8/12/16mA	-
	1	NCLE	O	CU,CD	4/8/12/16mA	-
	2	EXT_XCS2	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
NCEB1	0	GPIO44	IO	CU,CD	4/8/12/16mA	-
	1	NCEB1	O	CU,CD	4/8/12/16mA	-
	2	IDDIG	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
NCEB0	0	GPIO45	IO	CU,CD	4/8/12/16mA	-
	1	NCEB0	O	CU,CD	4/8/12/16mA	-
	2	DRV_VBUS	O	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
IR	0	GPIO46	IO	CU,CD	-	-
	1	IR	I	CU,CD	-	-
	2	-	-	CU,CD	-	-
	3	-	-	CU,CD	-	-
	4	-	-	CU,CD	-	-
	5	-	-	CU,CD	-	-
	6	-	-	CU,CD	-	-
	7	-	-	CU,CD	-	-
NREB	0	GPIO47	IO	CU,CD	4/8/12/16mA	-
	1	NREB	O	CU,CD	4/8/12/16mA	-
	2	IDDIG_P1	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
NRNB	0	GPIO48	IO	CU,CD	4/8/12/16mA	-
	1	NRNB	I	CU,CD	4/8/12/16mA	-
	2	DRV_VBUS_P1	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
I2S0_DATA	0	GPIO49	IO	CU,CD	4/8/12/16mA	-
	1	I2S0_DATA	IO	CU,CD	4/8/12/16mA	-
	2	I2S0_DATA_BYPS	O	CU,CD	4/8/12/16mA	-
	3	PCM_TX	O	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_I2S_DO	O	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[3]	IO	CU,CD	4/8/12/16mA	-
SPI0_CSN	0	GPIO53	IO	CU,CD	4/8/12/16mA	-
	1	SPI0_CS	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	SPDIF	O	CU,CD	4/8/12/16mA	-
	4	ADC_CK	O	CU,CD	4/8/12/16mA	-
	5	PWM1	O	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[7]	IO	CU,CD	4/8/12/16mA	-
SPI0_CK	0	GPIO54	IO	CU,CD	4/8/12/16mA	-
	1	SPI0_CK	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	SPDIF_IN1	I	CU,CD	4/8/12/16mA	-
	4	ADC_DAT_IN	I	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[10]	IO	CU,CD	4/8/12/16mA	-
SPI0_MI	0	GPIO55	IO	CU,CD	4/8/12/16mA	-
	1	SPI0_MI	I	CU,CD	4/8/12/16mA	-
	2	SPI0_MO	O	CU,CD	4/8/12/16mA	-
	3	MSDC1_WP	I	CU,CD	4/8/12/16mA	-
	4	ADC_WS	O	CU,CD	4/8/12/16mA	-
	5	PWM2	O	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[8]	IO	CU,CD	4/8/12/16mA	-
SPI0_MO	0	GPIO56	IO	CU,CD	4/8/12/16mA	-
	1	SPI0_MO	O	CU,CD	4/8/12/16mA	-
	2	SPI0_MI	I	CU,CD	4/8/12/16mA	-
	3	SPDIF_IN0	I	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[9]	IO	CU,CD	4/8/12/16mA	-
SDA1	0	GPIO57	IO	CD	-	-
	1	SDA1	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SCL1	0	GPIO58	IO	CD	-	-
	1	SCL1	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
WB_RSTB	0	GPIO60	IO	CU,CD	2/4/6/8mA	-
	1	WB_RSTB	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[11]	IO	CU,CD	2/4/6/8mA	-
GPIO61	0	GPIO61	IO	CU,CD	2/4/6/8mA	-
	1	-	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[16]	IO	CU,CD	2/4/6/8mA	-
GPIO62	0	GPIO62	IO	CU,CD	2/4/6/8mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	1	-	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[15]	IO	CU,CD	2/4/6/8mA	-
WB_SCLK	0	GPIO63	IO	CU,CD	2/4/6/8mA	-
	1	WB_SCLK	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[13]	IO	CU,CD	2/4/6/8mA	-
WB_SDATA	0	GPIO64	IO	CU,CD	2/4/6/8mA	-
	1	WB_SDATA	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[12]	IO	CU,CD	2/4/6/8mA	-
WB_SEN	0	GPIO65	IO	CU,CD	2/4/6/8mA	-
	1	WB_SEN	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[14]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL0	0	GPIO66	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL0	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	DFD_NTRST_XI	I	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[17]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL1	0	GPIO67	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL1	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	DFD_TMS_XI	I	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[18]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL2	0	GPIO68	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL2	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	4	-	-	CU,CD	2/4/6/8mA	-
	5	DFD_TCK_XI	I	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[19]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL3	0	GPIO69	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL3	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	DFD_TDI_XI	I	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[20]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL4	0	GPIO70	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL4	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	DFD_TDO	O	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[21]	IO	CU,CD	2/4/6/8mA	-
WB_CTRL5	0	GPIO71	IO	CU,CD	2/4/6/8mA	-
	1	WB_CTRL5	IO	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_A[22]	IO	CU,CD	2/4/6/8mA	-
I2S0_DATA_IN	0	GPIO72	IO	CU,CD	4/8/12/16mA	-
	1	I2S0_DATA_IN	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_RX	I	CU,CD	4/8/12/16mA	-
	4	PWM0	O	CU,CD	4/8/12/16mA	-
	5	DISP_PWM	O	CU,CD	4/8/12/16mA	-
	6	WCN_I2S_DI	I	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[2]	IO	CU,CD	4/8/12/16mA	-
I2S0_LRCK	0	GPIO73	IO	CU,CD	4/8/12/16mA	-
	1	I2S0_LRCK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_SYNC	IO	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_I2S_LRCK	IO	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[5]	IO	CU,CD	4/8/12/16mA	-
I2S0_BCK	0	GPIO74	IO	CU,CD	4/8/12/16mA	-
	1	I2S0_BCK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	PCM_CLK0	IO	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_I2S_BCK	IO	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	7	DBG_MON_B[4]	IO	CU,CD	4/8/12/16mA	-
SDA0	0	GPIO75	IO	CD	-	-
	1	SDA0	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SCL0	0	GPIO76	IO	CD	-	-
	1	SCL0	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SDA2	0	GPIO77	IO	CD	-	-
	1	SDA2	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SCL2	0	GPIO78	IO	CD	-	-
	1	SCL2	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
URXD0	0	GPIO79	IO	CD	-	-
	1	URXD0	I	CD	-	-
	2	UTXD0	O	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5		IO	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
UTXD0	0	GPIO80	IO	CD	-	-
	1	UTXD0	O	CD	-	-
	2	URXD0	I	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
URXD1	0	GPIO81	IO	CD	-	-
	1	URXD1	I	CD	-	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	2	UTXD1	O	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
UTXD1	0	GPIO82	IO	CD	-	-
	1	UTXD1	O	CD	-	-
	2	URXD1	I	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
LCM_RST	0	GPIO83	IO	CU,CD	2/4/6/8mA	-
	1	LCM_RST	O	CU,CD	2/4/6/8mA	-
	2	VDAC_CK_XI	I	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[1]	IO	CU,CD	2/4/6/8mA	-
DSI_TE	0	GPIO84	IO	CU,CD	2/4/6/8mA	-
	1	DSI_TE	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[0]	IO	CU,CD	2/4/6/8mA	-
	6	I2SOUT_LRCK	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	UTXD0	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	I2SOUT_DATA_OUT	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MIPI_TCN	0	GPIO95	IO	-	-	-
	1	MIPI_TCN	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
MIPI_TCP	0	GPIO96	IO	-	-	-
	1	MIPI_TCP	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
MIPI_TDN1	0	GPIO97	IO	-	-	-
	1	MIPI_TDN1	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
MIPI_TDP1	0	GPIO98	IO	-	-	-
	1	MIPI_TDP1	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
MIPI_TDN0	0	GPIO99	IO	-	-	-
	1	MIPI_TDN0	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
MIPI_TDP0	0	GPIO100	IO	-	-	-
	1	MIPI_TDP0	O	-	-	-
	2	-	-	-	-	-
	3	-	-	-	-	-
	4	-	-	-	-	-
	5	-	-	-	-	-
	6	-	-	-	-	-
	7	-	-	-	-	-
SPI2_CSN	0	GPIO101	IO	CD	-	-
	1	SPI2_CS	O	CD	-	-
	2	-	-	CD	-	-
	3	-	IO	CD	-	-
	4	KROW0	IO	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SPI2_MI	0	GPIO102	IO	CD	-	-
	1	SPI2_MI	I	CD	-	-
	2	SPI2_MO	O	CD	-	-
	3	-	IO	CD	-	-
	4	KROW1	IO	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	7	-	-	CD	-	-
SPI2_MO	0	GPIO103	IO	CD	-	-
	1	SPI2_MO	O	CD	-	-
	2	SPI2_MI	I	CD	-	-
	3	-	IO	CD	-	-
	4	KROW2	IO	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
SPI2_CK	0	GPIO104	IO	CD	-	-
	1	SPI2_CK	O	CD	-	-
	2	-	-	CD	-	-
	3	-	IO	CD	-	-
	4	KROW3	IO	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
MSDC1_CMD	0	GPIO105	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_CMD	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL0	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	SDA1	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	I2SOUT_BCK	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[27]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC1_CLK	0	GPIO106	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_CLK	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL1	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	SCL1	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	I2SOUT_LRCK	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[28]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC1_DAT0	0	GPIO107	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_DAT0	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL2	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	UTXD0	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
					3.3V: 4/8/12/16mA	
	6	I2SOUT_DATA_OUT	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[26]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC1_DAT1	0	GPIO108	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_DAT1	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL3	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	PWM0	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	URXD0	I	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	PWM1	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[25]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC1_DAT2	0	GPIO109	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_DAT2	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL4	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	SDA2	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	UTXD1	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	PWM2	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[24]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC1_DAT3	0	GPIO110	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC1_DAT3	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	ANT_SEL5	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	SCL2	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	EXT_FRAME_SYNC	I	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	URXD1	I	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	PWM3	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	DBG_MON_B[23]	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT7	0	GPIO111	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT7	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD7	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
					3.3V: 4/8/12/16mA	
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT6	0	GPIO112	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT6	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD6	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT5	0	GPIO113	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT5	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD5	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT4	0	GPIO114	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT4	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD4	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_RSTB	0	GPIO115	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_RSTB	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD8	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
					3.3V: 4/8/12/16mA	
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_CMD	0	GPIO116	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_CMD	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NALE	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_CLK	0	GPIO117	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_CLK	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NWEB	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT3	0	GPIO118	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT3	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD3	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT2	0	GPIO119	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT2	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD2	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
					3.3V: 4/8/12/16mA	
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT1	0	GPIO120	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT1	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD1	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
MSDC0_DAT0	0	GPIO121	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	1	MSDC0_DAT0	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	2	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	3	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	4	NLD0	IO	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	5	WATCHDOG	O	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	6	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
	7	-	-	CU,CD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	-
CEC	0	GPIO122	IO	CD	-	-
	1	CEC	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	SDA2	IO	CD	-	-
	5	URXD0	I	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
HTPLG	0	GPIO123	IO	CD	-	-
	1	HTPLG	I	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	SCL2	IO	CD	-	-
	5	UTXD0	O	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
HDMISCK	0	GPIO124	IO	CD	-	-
	1	HDMISCK	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	SDA1	IO	CD	-	-
	5	PWM3	O	CD	-	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	6	-	-	CD	-	-
	7	-	-	CD	-	-
HDMISD	0	GPIO125	IO	CD	-	-
	1	HDMISD	IO	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	SCL1	IO	CD	-	-
	5	PWM4	O	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
I2S0_MCLK	0	GPIO126	IO	CU,CD	4/8/12/16mA	-
	1	I2S0_MCLK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	WCN_I2S_MCLK	O	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[6]	IO	CU,CD	4/8/12/16mA	-
SPI1_CK	0	GPIO199	IO	CU,CD	2/4/6/8mA	-
	1	SPI1_CK	O	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	EXT_FRAME_SYNC	I	CU,CD	2/4/6/8mA	-
	4	KCOL3	IO	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	DBG_MON_B[15]	IO	CU,CD	2/4/6/8mA	-
SPDIF_OUT	0	GPIO200	IO	CU,CD	4/8/12/16mA	-
	1	SPDIF_OUT	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	URXD2	I	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[16]	IO	CU,CD	4/8/12/16mA	-
SPDIF_IN0	0	GPIO201	IO	CU,CD	4/8/12/16mA	-
	1	SPDIF_IN0	I	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	UTXD2	O	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[17]	IO	CU,CD	4/8/12/16mA	-
SPDIF_IN1	0	GPIO202	IO	CU,CD	4/8/12/16mA	-
	1	SPDIF_IN1	I	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
PWM0	0	GPIO203	IO	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	1	PWM0	O	CU,CD	4/8/12/16mA	-
	2	DISP_PWM	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[18]	IO	CU,CD	4/8/12/16mA	-
PWM1	0	GPIO204	IO	CU,CD	4/8/12/16mA	-
	1	PWM1	O	CU,CD	4/8/12/16mA	-
	2	CLKM3	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[19]	IO	CU,CD	4/8/12/16mA	-
PWM2	0	GPIO205	IO	CU,CD	4/8/12/16mA	-
	1	PWM2	O	CU,CD	4/8/12/16mA	-
	2	CLKM2	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[20]	IO	CU,CD	4/8/12/16mA	-
PWM3	0	GPIO206	IO	CU,CD	4/8/12/16mA	-
	1	PWM3	O	CU,CD	4/8/12/16mA	-
	2	CLKM1	O	CU,CD	4/8/12/16mA	-
	3	EXT_FRAME_SYNC	I	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[21]	IO	CU,CD	4/8/12/16mA	-
PWM4	0	GPIO207	IO	CU,CD	4/8/12/16mA	-
	1	PWM4	O	CU,CD	4/8/12/16mA	-
	2	CLKM0	O	CU,CD	4/8/12/16mA	-
	3	EXT_FRAME_SYNC	I	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[22]	IO	CU,CD	4/8/12/16mA	-
AUD_EXT_CK1	0	GPIO208	IO	CU,CD	4/8/12/16mA	-
	1	AUD_EXT_CK1	I	CU,CD	4/8/12/16mA	-
	2	PWM0	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	ANT_SEL5	O	CU,CD	4/8/12/16mA	-
	5	DISP_PWM	O	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[31]	IO	CU,CD	4/8/12/16mA	-
AUD_EXT_CK2	0	GPIO209	IO	CU,CD	4/8/12/16mA	-
	1	AUD_EXT_CK2	I	CU,CD	4/8/12/16mA	-
	2	MSDC1_WP	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	4	-	-	CU,CD	4/8/12/16mA	-
	5	PWM1	O	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[32]	IO	CU,CD	4/8/12/16mA	-
SFLASH_IO_3	0	GPIO236	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_IO_3	IO	CU,CD	4/8/12/16mA	-
	2	IDDIG	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[1]	IO	CU,CD	4/8/12/16mA	-
SFLASH_IO_2	0	GPIO237	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_IO_2	IO	CU,CD	4/8/12/16mA	-
	2	DRV_VBUS	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
SFLASH_IO_1	0	GPIO238	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_IO_1	IO	CU,CD	4/8/12/16mA	-
	2	IDDIG_P1	I	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
SFLASH_IO_0	0	GPIO239	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_IO_0	IO	CU,CD	4/8/12/16mA	-
	2	DRV_VBUS_P1	O	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
SFLASH_CS_L	0	GPIO240	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_CS_L	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
SFLASH_CLK	0	GPIO241	IO	CU,CD	4/8/12/16mA	-
	1	SFLASH_CLK	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	7	-	-	CU,CD	4/8/12/16mA	-
URTS2	0	GPIO242	IO	CU,CD	4/8/12/16mA	-
	1	URTS2	O	CU,CD	4/8/12/16mA	-
	2	UTXD3	O	CU,CD	4/8/12/16mA	-
	3	URXD3	I	CU,CD	4/8/12/16mA	-
	4	SCL1	IO	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_B[32]	IO	CU,CD	4/8/12/16mA	-
UCTS2	0	GPIO243	IO	CU,CD	4/8/12/16mA	-
	1	UCTS2	I	CU,CD	4/8/12/16mA	-
	2	URXD3	I	CU,CD	4/8/12/16mA	-
	3	UTXD3	O	CU,CD	4/8/12/16mA	-
	4	SDA1	IO	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	DBG_MON_A[6]	IO	CU,CD	4/8/12/16mA	-
GPIO244	0	GPIO244	IO	CD	-	-
	1	-	-	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
GPIO245	0	GPIO245	IO	CD	-	-
	1	-	-	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
MHL_SENCE	0	GPIO246	IO	CD	-	-
	1	-	-	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
GPIO247	0	GPIO247	IO	CD	-	-
	1	-	-	CD	-	-
	2	-	-	CD	-	-
	3	-	-	CD	-	-
	4	-	-	CD	-	-
	5	-	-	CD	-	-
	6	-	-	CD	-	-
	7	-	-	CD	-	-
GPIO248	0	GPIO248	IO	CU,CD	4/8/12/16mA	-
	1	HDMI_TESTOUTP_RX	O	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GPIO250	0	GPIO250	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO251	0	GPIO251	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO252	0	GPIO252	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO253	0	GPIO253	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO254	0	GPIO254	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO255	0	GPIO255	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO256	0	GPIO256	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GPIO257	0	GPIO257	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	1	-	IO	CU,CD	2/4/6/8/10/12/14/16mA	-
	2	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	3	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	4	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	5	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	6	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
	7	-	-	CU,CD	2/4/6/8/10/12/14/16mA	-
GE2_TXEN	0	GPIO262	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXEN	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_TXD3	0	GPIO263	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXD3	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL5	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_TXD2	0	GPIO264	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXD2	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL4	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_TXD1	0	GPIO265	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXD1	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL3	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
GE2_TXD0	0	GPIO266	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXD0	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL2	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_TXCLK	0	GPIO267	IO	CU,CD	4/8/12/16mA	-
	1	GE2_TXCLK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXCLK	0	GPIO268	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXCLK	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXD0	0	GPIO269	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXD0	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXD1	0	GPIO270	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXD1	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXD2	0	GPIO271	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXD2	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXD3	0	GPIO272	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXD3	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-

Name	Aux. function	Aux.name	Aux. type	PU/PD/ CU/CD	Driving	SMT
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
GE2_RXDV	0	GPIO274	IO	CU,CD	4/8/12/16mA	-
	1	GE2_RXDV	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	-	-	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
MDC	0	GPIO275	IO	CU,CD	4/8/12/16mA	-
	1	MDC	O	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL0	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
MDIO	0	GPIO276	IO	CU,CD	4/8/12/16mA	-
	1	MDIO	IO	CU,CD	4/8/12/16mA	-
	2	-	-	CU,CD	4/8/12/16mA	-
	3	-	-	CU,CD	4/8/12/16mA	-
	4	-	-	CU,CD	4/8/12/16mA	-
	5	-	-	CU,CD	4/8/12/16mA	-
	6	ANT_SEL1	O	CU,CD	4/8/12/16mA	-
	7	-	-	CU,CD	4/8/12/16mA	-
JTAG_RESET	0	GPIO278	IO	CU,CD	2/4/6/8mA	-
	1	JTAG_RESET	I	CU,CD	2/4/6/8mA	-
	2	-	-	CU,CD	2/4/6/8mA	-
	3	-	-	CU,CD	2/4/6/8mA	-
	4	-	-	CU,CD	2/4/6/8mA	-
	5	-	-	CU,CD	2/4/6/8mA	-
	6	-	-	CU,CD	2/4/6/8mA	-
	7	-	-	CU,CD	2/4/6/8mA	-

Below table shows the reset status of GPIOs.

Table 1-2. Reset Status of GPIOs

Name	Reset			Output drivability	Termination when not used	IO type
	State	Aux	PU/PD			
SYSRSTB	I	-	PU	-	No need	IO type 1
WATCHDOG	OH	1	PD	2/4/6/8mA	No need	IO type 2
TESTMODE	I	-	PD	-	No need	IO type 1
RTC32K_CK	I	1	PD	2/4/6/8mA	No need	IO type 2

SRCLKENAI	I	1	PD	2/4/6/8mA	No need	IO type 2
SRCLKENA	OH	1	PU	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_MO	OL	1	PD	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_MI	OL	1	PD	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_CSN	OH	1	PU	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_CK	OL	1	PD	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_CK2	OL	1	PD	2/4/6/8mA	No need	IO type 2
PWRAP_SPI0_CSN2	OH	1	PU	2/4/6/8mA	No need	IO type 2
PWRAP_INT	I	0	PD	2/4/6/8mA	No need	IO type 2
JTMS	I	1	PU	2/4/6/8mA	No need	IO type 2
JTCK	I	1	PU	2/4/6/8mA	No need	IO type 2
JTDI	I	1	PU	2/4/6/8mA	No need	IO type 2
JTDO	OL	1	PU	2/4/6/8mA	No need	IO type 2
JTAG_RESET	I	0	PU	2/4/6/8mA	No need	IO type 2
DSI_TE	I	0	PD	2/4/6/8mA	No need	IO type 2
LCM_RST	I	0	PD	2/4/6/8mA	No need	IO type 2
I2S0_DATA	OL	1	PD	4/8/12/16mA	No need	IO type 2
I2S0_DATA_IN	I	1	PD	4/8/12/16mA	No need	IO type 2
I2S0_LRCK	OH	1	PD	4/8/12/16mA	No need	IO type 2
I2S0_MCLK	OL	1	PD	4/8/12/16mA	No need	IO type 2
I2S0_BCK	OH	1	PD	4/8/12/16mA	No need	IO type 2
I2S1_DATA	OL	1	PD	4/8/12/16mA	No need	IO type 2
I2S1_DATA_IN	I	1	PD	4/8/12/16mA	No need	IO type 2
I2S1_LRCK	OH	1	PD	4/8/12/16mA	No need	IO type 2
I2S1_MCLK	OL	1	PD	4/8/12/16mA	No need	IO type 2
I2S1_BCK	OH	1	PD	4/8/12/16mA	No need	IO type 2
PCM_CLK	I	0	PD	4/8/12/16mA	No need	IO type 2
PCM_SYNC	I	0	PD	4/8/12/16mA	No need	IO type 2
PCM_RX	I	0	PD	4/8/12/16mA	No need	IO type 2
PCM_TX	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT0	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT1	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT2	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT3	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT4	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT5	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT6	I	0	PD	4/8/12/16mA	No need	IO type 2
EINT7	I	1	PD	4/8/12/16mA	No need	IO type 2
URXD0	I	1	-	-	No need	IO type 2
UTXD0	OH	1	-	-	No need	IO type 2

URXD1	I	0	PD	-	No need	IO type 2
UTXD1	I	0	PD	-	No need	IO type 2
URXD2	I	0	PD	-	No need	IO type 2
UTXD2	I	0	PD	-	No need	IO type 2
URTS2	I	0	PD	4/8/12/16mA	No need	IO type 2
UCTS2	I	0	PD	4/8/12/16mA	No need	IO type 2
SPI0_CSN	I	0	PD	4/8/12/16mA	No need	IO type 2
SPI0_CK	I	0	PD	4/8/12/16mA	No need	IO type 2
SPI0_MI	I	0	PD	4/8/12/16mA	No need	IO type 2
SPI0_MO	I	0	PD	4/8/12/16mA	No need	IO type 2
SPI1_CSN	I	0	PD	2/4/6/8mA	No need	IO type 2
SPI1_CK	I	0	PD	2/4/6/8mA	No need	IO type 2
SPI1_MI	I	0	PD	2/4/6/8mA	No need	IO type 2
SPI1_MO	I	0	PD	2/4/6/8mA	No need	IO type 2
SPI2_CSN	I	0	PD	-	No need	IO type 2
SPI2_CK	I	0	PD	-	No need	IO type 2
SPI2_MI	I	0	PD	-	No need	IO type 2
SPI2_MO	I	0	PD	-	No need	IO type 2
MSDC0_DAT7	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT6	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT5	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT4	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_RSTB	OH	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC0_CMD	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_CLK	OL	1	PD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT3	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT2	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT1	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC0_DAT0	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	GND	IO type 2
MSDC1_CMD	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC1_CLK	OL	1	PD	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC1_DAT0	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC1_DAT1	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC1_DAT2	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
MSDC1_DAT3	I	1	PU	1.8V: 2/4/6/8/10/12/14/16mA 3.3V: 4/8/12/16mA	No need	IO type 2
GPIO250	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2

GPIO251	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO252	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO253	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO254	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO255	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO256	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO257	I	1	PU	2/4/6/8/10/12/14/16mA	GND	IO type 2
GPIO61	I	-	PD	2/4/6/8mA	No need	IO type 2
GPIO62	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_SCLK	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_SDATA	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_SEN	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL0	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL1	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL2	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL3	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL4	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_CTRL5	I	-	PD	2/4/6/8mA	No need	IO type 2
WB_RSTB	I	-	PD	2/4/6/8mA	No need	IO type 2
SDA0	I	1	-	-	No need	Floating Well
SCL0	I	1	-	-	No need	Floating Well
SDA1	I	1	-	-	No need	Floating Well
SCL1	I	1	-	-	No need	Floating Well
SDA2	I	1	-	-	No need	Floating Well
SCL2	I	1	-	-	No need	Floating Well
CEC	I	1	-	-	No need	Floating Well
HTPLG	I	1	-	-	No need	Floating Well
HDMISCK	I	1	-	-	No need	Floating Well
HDMISD	I	1	-	-	No need	Floating Well
GPIO244	I	1	-	-	No need	Floating Well
GPIO245	I	1	-	-	No need	Floating Well
MHL_SENCE	I	0	-	-	No need	Floating Well
GPIO247	I	1	-	-	No need	Floating Well
GPIO248	OL	1	-	-	No need	IO type 2
MIPI_TDP0	I	1	-	-	No need	MIPI
MIPI_TDN0	I	1	-	-	No need	MIPI
MIPI_TDP1	I	1	-	-	No need	MIPI

MIPI_TDN1	I	1	-	-	No need	MIPI
MIPI_TCP	I	1	-	-	No need	MIPI
MIPI_TCN	I	1	-	-	No need	MIPI
NCLE	I	0	PD	4/8/12/16mA	No need	IO type 2
NCEB1	I	0	PU	4/8/12/16mA	No need	IO type 2
NCEB0	I	0	PU	4/8/12/16mA	No need	IO type 2
NREB	I	0	PD	4/8/12/16mA	No need	IO type 2
NRNB	I	0	PU	4/8/12/16mA	No need	IO type 2
PWM0	I	0	PD	4/8/12/16mA	No need	IO type 2
PWM1	I	0	PD	4/8/12/16mA	No need	IO type 2
PWM2	I	0	PD	4/8/12/16mA	No need	IO type 2
PWM3	I	0	PD	4/8/12/16mA	No need	IO type 2
PWM4	I	0	PD	4/8/12/16mA	No need	IO type 2
IR	I	0	PD	-	No need	IO type 1
SPDIF_OUT	I	0	PD	4/8/12/16mA	No need	IO type 2
SPDIF_IN0	I	0	PD	4/8/12/16mA	No need	IO type 2
SPDIF_IN1	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_IO_3	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_IO_2	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_IO_1	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_IO_0	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_CS_L	I	0	PD	4/8/12/16mA	No need	IO type 2
SFLASH_CLK	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXD3	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXD2	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXD1	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXD0	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXD3	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXD2	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXD1	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXD0	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXCLK	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXCLK	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_RXDV	I	0	PD	4/8/12/16mA	No need	IO type 2
GE2_TXEN	I	0	PD	4/8/12/16mA	No need	IO type 2
MDC	I	0	PD	4/8/12/16mA	No need	IO type 2
MDIO	I	0	PD	4/8/12/16mA	No need	IO type 2
AUD_EXT_CK1	I	1	PD	4/8/12/16mA	No need	IO type 2
AUD_EXT_CK2	I	1	PD	4/8/12/16mA	No need	IO type 2

Below table shows the driving control register name and steps.

Table 1-3: Driving and pull control of GPIOs

Name	Mode 0	IES/ SMT	OCTL	Pull type	Pull init	Pull_en	Pull_sel	Pu Pd	PuPd/ R1/R0	Driving register	Driving step
PWRAP_SPI0_MI	GPIO0	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PD	0x10005150[0]	0x10005280[0]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_SPI0_MO	GPIO1	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PD	0x10005150[1]	0x10005280[1]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_INT	GPIO2	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PD	0x10005150[2]	0x10005280[2]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_SPI0_CK	GPIO3	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PD	0x10005150[3]	0x10005280[3]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_SPI0_CSN	GPIO4	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PU	0x10005150[4]	0x10005280[4]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_SPI0_CK2	GPIO5	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PD	0x10005150[5]	0x10005280[5]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWRAP_SPI0_CSN2	GPIO6	IES0 SMT0	pwrap_conf[3:0]	PU/PD	PU	0x10005150[6]	0x10005280[6]	-	-	0x10005F50[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
SPI1_CSN	GPIO7	IES1 SMT1	spi1_conf[3:0]	PU/PD	PD	0x10005150[7]	0x10005280[7]	-	-	0x10005F50[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SPI1_MI	GPIO8	IES1 SMT1	spi1_conf[3:0]	PU/PD	PD	0x10005150[8]	0x10005280[8]	-	-	0x10005F50[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SPI1_MO	GPIO9	IES1 SMT1	spi1_conf[3:0]	PU/PD	PD	0x10005150[9]	0x10005280[9]	-	-	0x10005F50[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
RTC32K_CK	GPIO10	IES2 SMT2	pm_conf[3:0]	PU/PD	PD	0x10005150[10]	0x10005280[10]	-	-	0x10005F50[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
WATCHDOG	GPIO11	IES2 SMT2	pm_conf[3:0]	PU/PD	PU	0x10005150[11]	0x10005280[11]	-	-	0x10005F50[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
SRCLKENA	GPIO12	IES2 SMT2	pm_conf[3:0]	PU/PD	PU	0x10005150[12]	0x10005280[12]	-	-	0x10005F50[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
SRCLKENAI	GPIO13	IES2 SMT2	pm_conf[3:0]	PU/PD	PD	0x10005150[13]	0x10005280[13]	-	-	0x10005F50[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
URXD2	GPIO14	IES3 SMT3	uart2_conf[3:0]	PU/PD	PD	0x10005150[14]	0x10005280[14]	-	-	0x10005F50[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
UTXD2	GPIO15	IES3 SMT3	uart2_conf[3:0]	PU/PD	PD	0x10005150[15]	0x10005280[15]	-	-	0x10005F50[15:12]	0x00000000,0x00002000,0x00004000,0x00006000

		SMT4									
PCM_CLK	GPIO18	IES5 SMT5	pcm_conf[3:0]	PU/PD	PD	0x10005160[2]	0x10005290[2]	-	-	0x10005F60[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PCM_SYNC	GPIO19	IES5 SMT5	pcm_conf[3:0]	PU/PD	PD	0x10005160[3]	0x10005290[3]	-	-	0x10005F60[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PCM_RX	GPIO20	IES5 SMT5	pcm_conf[3:0]	PU/PD	PD	0x10005160[4]	0x10005290[4]	-	-	0x10005F60[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PCM_TX	GPIO21	IES5 SMT5	pcm_conf[3:0]	PU/PD	PD	0x10005160[5]	0x10005290[5]	-	-	0x10005F60[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
EINT0	GPIO22	IES6 SMT6	eint0_conf[3:0]	PU/PD	PD	0x10005160[6]	0x10005290[6]	-	-	0x10005F60[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
EINT1	GPIO23	IES6 SMT6	eint0_conf[3:0]	PU/PD	PD	0x10005160[7]	0x10005290[7]	-	-	0x10005F60[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
EINT2	GPIO24	IES6 SMT6	eint0_conf[3:0]	PU/PD	PD	0x10005160[8]	0x10005290[8]	-	-	0x10005F60[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
EINT3	GPIO25	IES6 SMT6	eint0_conf[3:0]	PU/PD	PD	0x10005160[9]	0x10005290[9]	-	-	0x10005F60[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
EINT4	GPIO26	IES6 SMT6	eint0_conf[3:0]	PU/PD	PD	0x10005160[10]	0x10005290[10]	-	-	0x10005F60[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
EINT5	GPIO27	IES7 SMT7	eint5_conf[3:0]	PU/PD	PD	0x10005160[11]	0x10005290[11]	-	-	0x10005F60[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
EINT6	GPIO28	IES7 SMT7	eint5_conf[3:0]	PU/PD	PD	0x10005160[12]	0x10005290[12]	-	-	0x10005F60[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
EINT7	GPIO29	IES7 SMT7	eint5_conf[3:0]	PU/PD	PD	0x10005160[13]	0x10005290[13]	-	-	0x10005F60[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
I2S1_DATA	GPIO33	IES8 SMT8	i2s1_conf[3:0]	PU/PD	PD	0x10005170[1]	0x10005 2A0[1]	-	-	0x10005F70[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S1_DATA_IN	GPIO34	IES8 SMT8	i2s1_conf[3:0]	PU/PD	PD	0x10005170[2]	0x10005 2A0[2]	-	-	0x10005F70[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S1_BCK	GPIO35	IES8 SMT8	i2s1_conf[3:0]	PU/PD	PD	0x10005170[3]	0x10005 2A0[3]	-	-	0x10005F70[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S1_LRCK	GPIO36	IES8 SMT8	i2s1_conf[3:0]	PU/PD	PD	0x10005170[4]	0x10005 2A0[4]	-	-	0x10005F70[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S1_MCLK	GPIO37	IES8 SMT8	i2s1_conf[3:0]	PU/PD	PD	0x10005170[5]	0x10005 2A0[5]	-	-	0x10005F70[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
JTMS	GPIO39	IES9 SMT9	jtag_conf[3:0]	PU/PD	PU	0x10005170[7]	0x10005 2A0[7]	-	-	0x10005F70[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
JTCK	GPIO4	IES9	jtag_conf	PU/P	PU	0x10005	0x10005	-	-	0x10005F7	0x00000000,0x00000200,0x000004

	0	SMT9	[3:0]	D		170[8]	2A0[8]			0[11:8]	00,0x00000600
JTDI	GPIO41	IES9 SMT9	jtag_conf[3:0]	PU/PD	PU	0x10005170[9]	0x100052A0[9]	-	-	0x10005F70[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
JTDO	GPIO42	IES9 SMT9	jtag_conf[3:0]	PU/PD	PU	0x10005170[10]	0x100052A0[10]	-	-	0x10005F70[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
NCLE	GPIO43	IES10 SMT10	eint12_conf[3:0]	PU/PD	PD	0x10005170[11]	0x100052A0[11]	-	-	0x10005F70[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
NCEB1	GPIO44	IES10 SMT10	eint12_conf[3:0]	PU/PD	PU	0x10005170[12]	0x100052A0[12]	-	-	0x10005F70[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
NCEB0	GPIO45	IES10 SMT10	eint12_conf[3:0]	PU/PD	PU	0x10005170[13]	0x100052A0[13]	-	-	0x10005F70[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
IR	GPIO46	1'b1 1'b0	-	PU/PD	PD	0x10005170[14]	0x100052A0[14]	-	-	-	-
NREB	GPIO47	IES11 SMT11	eint17_conf[3:0]	PU/PD	PD	0x10005170[15]	0x100052A0[15]	-	-	0x10005F80[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
NRNB	GPIO48	IES11 SMT11	eint17_conf[3:0]	PU/PD	PU	0x10005180[0]	0x100052B0[0]	-	-	0x10005F80[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S0_DATA	GPIO49	IES12 SMT12	i2s0_conf[3:0]	PU/PD	PD	0x10005180[1]	0x100052B0[1]	-	-	0x10005F80[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SPI0_CSN	GPIO53	IES14 SMT14	spi0_conf[3:0]	PU/PD	PD	0x10005180[5]	0x100052B0[5]	-	-	0x10005F80[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
SPI0_CK	GPIO54	IES14 SMT14	spi0_conf[3:0]	PU/PD	PD	0x10005180[6]	0x100052B0[6]	-	-	0x10005F80[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
SPI0_MI	GPIO55	IES14 SMT14	spi0_conf[3:0]	PU/PD	PD	0x10005180[7]	0x100052B0[7]	-	-	0x10005F80[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
SPI0_MO	GPIO56	IES14 SMT14	spi0_conf[3:0]	PU/PD	PD	0x10005180[8]	0x100052B0[8]	-	-	0x10005F80[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
SDA1	GPIO57	IES15 SMT15	-	PD	-	0x10005180[9]	0x100052B0[9]	-	-	-	-
SCL1	GPIO58	IES15 SMT15	-	PD	-	0x10005180[10]	0x100052B0[10]	-	-	-	-
WB_RSTB	GPIO60	IES16 SMT16	fm_conf[3:0]	PU/PD	PD	0x10005180[12]	0x100052B0[12]	-	-	0x10005F90[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GPIO61	GPIO61	IES16 SMT16	fm_conf[3:0]	PU/PD	PD	0x10005180[13]	0x100052B0[13]	-	-	0x10005F90[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GPIO62	GPIO62	IES16 SMT16	fm_conf[3:0]	PU/PD	PD	0x10005180[14]	0x100052B0[14]	-	-	0x10005F90[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
WB_SCLK	GPIO6	IES17	wbspi_co	PU/P	PD	0x10005	0x10005	-	-	0x10005F9	0x00000000,0x00002000,0x000040

	3	SMT17	nf[3:0]	D		180[15]	2B0[15]			0[15:12]	00,0x00006000
WB_SDATA	GPIO64	IES17 SMT17	wbspi_conf[3:0]	PU/PD	PD	0x10005190[0]	0x100052C0[0]	-	-	0x10005F90[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
WB_SEN	GPIO65	IES17 SMT17	wbspi_conf[3:0]	PU/PD	PD	0x10005190[1]	0x100052C0[1]	-	-	0x10005F90[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
WB_CTRL0	GPIO66	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[2]	0x100052C0[2]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
WB_CTRL1	GPIO67	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[3]	0x100052C0[3]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
WB_CTRL2	GPIO68	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[4]	0x100052C0[4]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
WB_CTRL3	GPIO69	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[5]	0x100052C0[5]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
WB_CTRL4	GPIO70	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[6]	0x100052C0[6]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
WB_CTRL5	GPIO71	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x10005190[7]	0x100052C0[7]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
I2S0_DATA_IN	GPIO72	IES12 SMT12	i2s0_conf[3:0]	PU/PD	PD	0x10005190[8]	0x100052C0[8]	-	-	0x10005F80[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
I2S0_LRCK	GPIO73	IES12 SMT12	i2s0_conf[3:0]	PU/PD	PD	0x10005190[9]	0x100052C0[9]	-	-	0x10005F80[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
I2S0_BCK	GPIO74	IES12 SMT12	i2s0_conf[3:0]	PU/PD	PD	0x10005190[10]	0x100052C0[10]	-	-	0x10005F80[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SDA0	GPIO75	IES19 SMT19	-	PD	-	0x10005190[11]	0x100052C0[11]	-	-	-	-
SCL0	GPIO76	IES19 SMT19	-	PD	-	0x10005190[12]	0x100052C0[12]	-	-	-	-
SDA2	GPIO77	IES20 SMT20	-	PD	-	0x10005190[13]	0x100052C0[13]	-	-	-	-
SCL2	GPIO78	IES20 SMT20	-	PD	-	0x10005190[14]	0x100052C0[14]	-	-	-	-
URXD0	GPIO79	IES21 SMT21	-	PD	-	0x10005190[15]	0x100052C0[15]	-	-	-	-
UTXD0	GPIO80	IES21 SMT21	-	PD	-	0x100051A0[0]	0x100052D0[0]	-	-	-	-
URXD1	GPIO81	IES21 SMT21	-	PD	-	0x100051A0[1]	0x100052D0[1]	-	-	-	-
UTXD1	GPIO8	IES21	-	PD	-	0x10005	0x10005	-	-	-	-

	2	SMT21				1A0[2]	2D0[2]				
LCM_RST	GPIO83	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x100051A0[3]	0x100052D0[3]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
DSI_TE	GPIO84	IES18 SMT18	wb_conf[3:0]	PU/PD	PD	0x100051A0[4]	0x100052D0[4]	-	-	0x10005FA0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
MIPI_TCN	GPI95	- /-	-	NP	NP	-	-	-	-	-	-
MIPI_TCP	GPI96	- /-	-	NP	NP	-	-	-	-	-	-
MIPI_TDN1	GPI97	- /-	-	NP	NP	-	-	-	-	-	-
MIPI_TDP1	GPI98	- /-	-	NP	NP	-	-	-	-	-	-
MIPI_TDN0	GPI99	- /-	-	NP	NP	-	-	-	-	-	-
MIPI_TDP0	GPI100	- /-	-	NP	NP	-	-	-	-	-	-
SPI2_CSN	GPIO101	IES22 SMT22	-	PD	-	0x100051B0[5]	0x100052E0[5]	-	-	-	-
SPI2_MI	GPIO102	IES22 SMT22	-	PD	-	0x100051B0[6]	0x100052E0[6]	-	-	-	-
SPI2_MO	GPIO103	IES22 SMT22	-	PD	-	0x100051B0[7]	0x100052E0[7]	-	-	-	-
SPI2_CK	GPIO104	IES22 SMT22	-	PD	-	0x100051B0[8]	0x100052E0[8]	-	-	-	-
MSDC1_CMD	GPIO105	0x100005D40[4] /0x100005D40[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D40[10:8]	0x10005D40[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC1_CLK	GPIO106	0x100005D30[4] /0x100005D30[11]	-	PU/PD	PD	-	-	-	{r0,r1,pupd}=0x10005D30[10:8]	0x10005D30[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC1_DAT0	GPIO107	0x100005D50[4] /0x100005D60[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D60[2:0]	0x10005D50[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC1_DAT1	GPIO108	0x100005D50[4] /0x100005D60[7]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D60[6:4]	0x10005D50[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,

MSDC1_DAT2	GPIO109	0x10005D50[4] /0x10005D60[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D60[10:8]	0x10005D50[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC1_DAT3	GPIO110	0x10005D50[4] /0x10005D60[15]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D60[14:12]	0x10005D50[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT7	GPIO111	0x10005CE0[4] /0x10005D00[15]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D00[14:12]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT6	GPIO112	0x10005CE0[4] /0x10005D00[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D00[10:8]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT5	GPIO113	0x10005CE0[4] /0x10005D00[7]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D00[6:4]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT4	GPIO114	0x10005CE0[4] /0x10005D00[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D00[2:0]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_RSTB	GPIO115	0x10005CE0[4] /0x10005D10[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005D10[2:0]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_CMD	GPIO116	0x10005CD0[4] /0x10005CD0[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005CD0[10:8]	0x10005CD0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_CLK	GPIO117	0x10005CC0[4] /0x10005CC0[11]	-	PU/PD	PD	-	-	-	{r0,r1,pupd}=0x10005CC0[10:8]	0x10005CC0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT3	GPIO118	0x10005CE0[4] /0x10005CF0[15]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005CF0[14:12]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,

MSDC0_DAT2	GPIO119	0x10005CE0[4] /0x10005CF0[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005CF0[10:8]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT1	GPIO120	0x10005CE0[4] /0x10005CF0[7]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005CF0[6:4]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
MSDC0_DAT0	GPIO121	0x10005CE0[4] /0x10005CF0[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005CF0[2:0]	0x10005CE0[3:0]	1.8V: 0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007- 3.3V: 0x00000000,0x00000001,0x00000002,0x00000003,
CEC	GPIO122	IES23 SMT23	-	PD	-	0x100051C0[10]	0x100052F0[10]	-	-	-	-
HTPLG	GPIO123	IES23 SMT23	-	PD	-	0x100051C0[11]	0x100052F0[11]	-	-	-	-
HDMISCK	GPIO124	IES23 SMT23	-	PD	-	0x100051C0[12]	0x100052F0[12]	-	-	-	-
HDMISD	GPIO125	IES23 SMT23	-	PD	-	0x100051C0[13]	0x100052F0[13]	-	-	-	-
I2S0_MCLK	GPIO126	IES12 SMT12	i2s0_conf[3:0]	PU/PD	PD	0x100051C0[14]	0x100052F0[14]	-	-	0x10005F80[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SPI1_CK	GPIO199	IES1 SMT1	spi1_conf[3:0]	PU/PD	PD	0x10005210[7]	0x10005340[7]	-	-	0x10005F50[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SPDIF_OUT	GPIO200	IES33 SMT33	spdif_conf[3:0]	PU/PD	PD	0x10005210[8]	0x10005340[8]	-	-	0x10005FD0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
SPDIF_IN0	GPIO201	IES33 SMT33	spdif_conf[3:0]	PU/PD	PD	0x10005210[9]	0x10005340[9]	-	-	0x10005FD0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
SPDIF_IN1	GPIO202	IES33 SMT33	spdif_conf[3:0]	PU/PD	PD	0x10005210[10]	0x10005340[10]	-	-	0x10005FD0[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
PWM0	GPIO203	IES34 SMT34	pwm_conf[3:0]	PU/PD	PD	0x10005210[11]	0x10005340[11]	-	-	0x10005FD0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PWM1	GPIO204	IES34 SMT34	pwm_conf[3:0]	PU/PD	PD	0x10005210[12]	0x10005340[12]	-	-	0x10005FD0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PWM2	GPIO205	IES34 SMT34	pwm_conf[3:0]	PU/PD	PD	0x10005210[13]	0x10005340[13]	-	-	0x10005FD0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PWM3	GPIO206	IES34 SMT34	pwm_conf[3:0]	PU/PD	PD	0x10005210[14]	0x10005340[14]	-	-	0x10005FD0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
PWM4	GPIO2	IES34	pwm_co	PU/P	PD	0x10005	0x10005	-	-	0x10005FD	0x00000000,0x00000020,0x000000

	07	SMT34	nf[3:0]	D		210[15]	340[15]			0[7:4]	40,0x00000060
AUD_EXT_CK1	GPIO208	IES35 SMT35	audck_conf[3:0]	PU/PD	PD	0x10005220[0]	0x10005350[0]	-	-	0x10005FD0[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
AUD_EXT_CK2	GPIO209	IES35 SMT35	audck_conf[3:0]	PU/PD	PD	0x10005220[1]	0x10005350[1]	-	-	0x10005FD0[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
SFLASH_IO_3	GPIO236	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005230[12]	0x10005360[12]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SFLASH_IO_2	GPIO237	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005230[13]	0x10005360[13]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SFLASH_IO_1	GPIO238	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005230[14]	0x10005360[14]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SFLASH_IO_0	GPIO239	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005230[15]	0x10005360[15]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SFLASH_CS_L	GPIO240	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005240[0]	0x10005370[0]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
SFLASH_CLK	GPIO241	IES38 SMT38	ext_conf[3:0]	PU/PD	PD	0x10005240[1]	0x10005370[1]	-	-	0x10005FF0[7:4]	0x00000000,0x00000020,0x00000040,0x00000060
URTS2	GPIO242	IES39 SMT39	ud_conf[3:0]	PU/PD	PD	0x10005240[2]	0x10005370[2]	-	-	0x10005FF0[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
UCTS2	GPIO243	IES39 SMT39	ud_conf[3:0]	PU/PD	PD	0x10005240[3]	0x10005370[3]	-	-	0x10005FF0[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GPIO244	GPIO244	IES40 SMT40	mhl_conf[3:0]	PD	-	0x10005240[4]	0x10005370[4]	-	-	0x10005FF0[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
GPIO245	GPIO245	IES40 SMT40	mhl_conf[3:0]	PD	-	0x10005240[5]	0x10005370[5]	-	-	0x10005FF0[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
MHL_SENCE	GPIO246	IES40 SMT40	mhl_conf[3:0]	PD	PD	0x10005240[6]	0x10005370[6]	-	-	0x10005FF0[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
GPIO247	GPIO247	IES40 SMT40	mhl_conf[3:0]	PD	PD	0x10005240[7]	0x10005370[7]	-	-	0x10005FF0[15:12]	0x00000000,0x00002000,0x00004000,0x00006000
GPIO248	GPIO248	IES41 SMT41	hdmiout_conf[3:0]	PD	-	0x10005240[8]	0x10005370[8]	-	-	0x10005F00[3:0]	0x00000000,0x00000002,0x00000004,0x00000006
GPIO250	GPIO250	0x10005FC0[4]/0x10005130[15]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005130[14:12]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO251	GPIO251	0x10005FC0[4]/0x10005130[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005130[10:8]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-

GPIO252	GPIO252	0x10005FC0[4]/0x10005130[7]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005130[6:4]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO253	GPIO253	0x10005FC0[4]/0x10005130[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005130[2:0]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO254	GPIO254	0x10005FC0[4]/0x10005F40[15]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005F40[14:12]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO255	GPIO255	0x10005FC0[4]/0x10005F40[11]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005F40[10:8]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO256	GPIO256	0x10005FC0[4]/0x10005F40[7]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005F40[6:4]	0x10005FC0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GPIO257	GPIO257	0x10005FC0[4]/0x10005F40[3]	-	PU/PD	PU	-	-	-	{r0,r1,pupd}=0x10005F40[2:0]	0x10005CE0[3:0]	0x00000000,0x00000001,0x00000002,0x00000003,0x00000004,0x00000005,0x00000006,0x00000007-
GE2_TXEN	GPIO262	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[6]	0x10005380[6]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_TXD3	GPIO263	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[7]	0x10005380[7]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_TXD2	GPIO264	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[8]	0x10005380[8]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_TXD1	GPIO265	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[9]	0x10005380[9]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_TXD0	GPIO266	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[10]	0x10005380[10]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_TXCLK	GPIO267	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[11]	0x10005380[11]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_RXCLK	GPIO268	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[12]	0x10005380[12]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_RXD0	GPIO269	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[13]	0x10005380[13]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_RXD1	GPIO270	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005250[14]	0x10005380[14]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_RXD	GPIO2	IES44	rgmii_co	PU/P	PD	0x10005	0x10005	-	-	0x10005F0	0x00000000,0x00000200,0x000004

2	71	SMT44	nf[3:0]	D		250[15]	380[15]			0[11:8]	00,0x00000600
GE2_RXD3	GPIO272	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005260[0]	0x10005390[0]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
GE2_RXDV	GPIO274	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005260[2]	0x10005390[2]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
MDC	GPIO275	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005260[3]	0x10005390[3]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
MDIO	GPIO276	IES44 SMT44	rgmii_conf[3:0]	PU/PD	PD	0x10005260[4]	0x10005390[4]	-	-	0x10005F00[11:8]	0x00000000,0x00000200,0x00000400,0x00000600
JTAG_RESET	GPIO278	IES45 SMT45	jtag_conf[3:0]	PU/PD	PU	0x10005260[6]	0x10005390[6]	-	-	0x10005F70[11:8]	0x00000000,0x00000200,0x00000400,0x00000600

Below Tables shows the drving control and SMT control registers. Refer to the datasheet of infrasys configuration for detail command registers.

Table 1-4: IES control register mapping

IES control bit	Register bit	Register bit
IES0	IES0_EN[0]	0x10005B20[0]
IES1	IES0_EN[1]	0x10005B20[1]
IES2	IES0_EN[2]	0x10005B20[2]
IES3	IES0_EN[3]	0x10005B20[3]
IES4	IES0_EN[4]	0x10005B20[4]
IES5	IES0_EN[5]	0x10005B20[5]
IES6	IES0_EN[6]	0x10005B20[6]
IES7	IES0_EN[7]	0x10005B20[7]
IES8	IES0_EN[8]	0x10005B20[8]
IES9	IES0_EN[9]	0x10005B20[9]
IES10	IES0_EN[10]	0x10005B20[10]
IES11	IES0_EN[11]	0x10005B20[11]
IES12	IES0_EN[12]	0x10005B20[12]
IES13	IES0_EN[13]	0x10005B20[13]
IES14	IES0_EN[14]	0x10005B20[14]
IES15	IES0_EN[15]	0x10005B20[15]
IES16	IES1_EN[0]	0x10005B30[0]
IES17	IES1_EN[1]	0x10005B30[1]
IES18	IES1_EN[2]	0x10005B30[2]
IES19	IES1_EN[3]	0x10005B30[3]
IES20	IES1_EN[4]	0x10005B30[4]
IES21	IES1_EN[5]	0x10005B30[5]
IES22	IES1_EN[6]	0x10005B30[6]

IES control bit	Register bit	Register bit
IES23	IES1_EN[7]	0x10005B30[7]
IES24	MSDC0_CTRL2[4]	0x10005CE0[4]
IES25	IES1_EN[9]	0x10005B30[9]
IES26	IES1_EN[10]	0x10005B30[10]
IES27	IES1_EN[11]	0x10005B30[11]
IES28	IES1_EN[12]	0x10005B30[12]
IES29	IES1_EN[13]	0x10005B30[13]
IES30	IES1_EN[14]	0x10005B30[14]
IES31	IES1_EN[15]	0x10005B30[15]
IES32	IES2_EN[0]	0x10005B40[0]
IES33	IES2_EN[1]	0x10005B40[1]
IES34	IES2_EN[2]	0x10005B40[2]
IES35	IES2_EN[3]	0x10005B40[3]
IES36	IES2_EN[4]	0x10005B40[4]
IES37	IES2_EN[5]	0x10005B40[5]
IES38	IES2_EN[6]	0x10005B40[6]
IES39	IES2_EN[7]	0x10005B40[7]
IES40	IES2_EN[8]	0x10005B40[8]
IES41	IES2_EN[9]	0x10005B40[9]
IES42	MSDC3_CTRL2[4]	0x10005FC0[4]
IES43	MSDC1_CTRL2[4]	0x10005D50[4]
IES44	IES2_EN[12]	0x10005B40[12]
IES45	IES2_EN[13]	0x10005B40[13]

Table 1-5: SMTcontrol register mapping

SMT control bit	Register bit	Register bit
SMT0	SMT0_EN[0]	0x10005B50[0]
SMT1	SMT0_EN[1]	0x10005B50[1]
SMT2	SMT0_EN[2]	0x10005B50[2]
SMT3	SMT0_EN[3]	0x10005B50[3]
SMT4	SMT0_EN[4]	0x10005B50[4]
SMT5	SMT0_EN[5]	0x10005B50[5]
SMT6	SMT0_EN[6]	0x10005B50[6]
SMT7	SMT0_EN[7]	0x10005B50[7]
SMT8	SMT0_EN[8]	0x10005B50[8]
SMT9	SMT0_EN[9]	0x10005B50[9]
SMT10	SMT0_EN[10]	0x10005B50[10]
SMT11	SMT0_EN[11]	0x10005B50[11]
SMT12	SMT0_EN[12]	0x10005B50[12]
SMT13	SMT0_EN[13]	0x10005B50[13]
SMT14	SMT0_EN[14]	0x10005B50[14]

SMT control bit	Register bit	Register bit
SMT15	SMT0_EN[15]	0x10005B50[15]
SMT16	SMT1_EN[0]	0x10005B60[0]
SMT17	SMT1_EN[1]	0x10005B60[1]
SMT18	SMT1_EN[2]	0x10005B60[2]
SMT19	SMT1_EN[3]	0x10005B60[3]
SMT20	SMT1_EN[4]	0x10005B60[4]
SMT21	SMT1_EN[5]	0x10005B60[5]
SMT22	SMT1_EN[6]	0x10005B60[6]
SMT23	SMT1_EN[7]	0x10005B60[7]
SMT24	MSDC0_CTRL5[3]	0x10005D10[3]
SMT25	SMT1_EN[9]	0x10005B60[9]
SMT26	SMT1_EN[10]	0x10005B60[10]
SMT27	SMT1_EN[11]	0x10005B60[11]
SMT28	SMT1_EN[12]	0x10005B60[12]
SMT29	SMT1_EN[13]	0x10005B60[13]
SMT30	SMT1_EN[14]	0x10005B60[14]
SMT31	SMT1_EN[15]	0x10005B60[15]
SMT32	SMT2_EN[0]	0x10005B70[0]
SMT33	SMT2_EN[1]	0x10005B70[1]
SMT34	SMT2_EN[2]	0x10005B70[2]
SMT35	SMT2_EN[3]	0x10005B70[3]
SMT36	SMT2_EN[4]	0x10005B70[4]
SMT37	SMT2_EN[5]	0x10005B70[5]
SMT38	SMT2_EN[6]	0x10005B70[6]
SMT39	SMT2_EN[7]	0x10005B70[7]
SMT40	SMT2_EN[8]	0x10005B70[8]
SMT41	SMT2_EN[9]	0x10005B70[9]
SMT42	MSDC3_CTRL5[3]	0x10005140[3]
SMT43	MSDC1_CTRL6[3]	0x100050B0[3]
SMT44	SMT2_EN[12]	0x10005B70[12]
SMT45	SMT2_EN[13]	0x10005B70[13]

Table 1-6: Driving control register mapping

Name	Driving control	Register name	MSB	LSB
PWRAP_SPI0_MI	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_SPI0_MO	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_INT	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_SPI0_CK	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_SPI0_CSN	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_SPI0_CK2	pwrap_conf[3:0]	DRV_SEL0	3	0
PWRAP_SPI0_CSN2	pwrap_conf[3:0]	DRV_SEL0	3	0

Name	Driving control	Register name	MSB	LSB
SPI1_CSN	spi1_conf[3:0]	DRV_SEL0	7	4
SPI1_MI	spi1_conf[3:0]	DRV_SEL0	7	4
SPI1_MO	spi1_conf[3:0]	DRV_SEL0	7	4
RTC32K_CK	pm_conf[3:0]	DRV_SEL0	11	8
WATCHDOG	pm_conf[3:0]	DRV_SEL0	11	8
SRCLKENA	pm_conf[3:0]	DRV_SEL0	11	8
SRCLKENAI	pm_conf[3:0]	DRV_SEL0	11	8
URXD2	uart2_conf[3:0]	DRV_SEL0	15	12
UTXD2	uart2_conf[3:0]	DRV_SEL0	15	12
PCM_CLK	pcm_conf[3:0]	DRV_SEL1	7	4
PCM_SYNC	pcm_conf[3:0]	DRV_SEL1	7	4
PCM_RX	pcm_conf[3:0]	DRV_SEL1	7	4
PCM_TX	pcm_conf[3:0]	DRV_SEL1	7	4
EINT0	eint0_conf[3:0]	DRV_SEL1	11	8
EINT1	eint0_conf[3:0]	DRV_SEL1	11	8
EINT2	eint0_conf[3:0]	DRV_SEL1	11	8
EINT3	eint0_conf[3:0]	DRV_SEL1	11	8
EINT4	eint0_conf[3:0]	DRV_SEL1	11	8
EINT5	eint5_conf[3:0]	DRV_SEL1	15	12
EINT6	eint5_conf[3:0]	DRV_SEL1	15	12
EINT7	eint5_conf[3:0]	DRV_SEL1	15	12
I2S1_DATA	i2s1_conf[3:0]	DRV_SEL2	3	0
I2S1_DATA_IN	i2s1_conf[3:0]	DRV_SEL2	3	0
I2S1_BCK	i2s1_conf[3:0]	DRV_SEL2	3	0
I2S1_LRCK	i2s1_conf[3:0]	DRV_SEL2	3	0
I2S1_MCLK	i2s1_conf[3:0]	DRV_SEL2	3	0
JTMS	jtag_conf[3:0]	DRV_SEL2	11	8
JTCK	jtag_conf[3:0]	DRV_SEL2	11	8
JTDI	jtag_conf[3:0]	DRV_SEL2	11	8
JTDO	jtag_conf[3:0]	DRV_SEL2	11	8
NCLE	eint12_conf[3:0]	DRV_SEL2	15	12
NCEB1	eint12_conf[3:0]	DRV_SEL2	15	12
NCEB0	eint12_conf[3:0]	DRV_SEL2	15	12
NREB	eint17_conf[3:0]	DRV_SEL3	3	0
NRNB	eint17_conf[3:0]	DRV_SEL3	3	0
I2S0_DATA	i2s0_conf[3:0]	DRV_SEL3	7	4
SPI0_CSN	spi0_conf[3:0]	DRV_SEL3	15	12
SPI0_CK	spi0_conf[3:0]	DRV_SEL3	15	12
SPI0_MI	spi0_conf[3:0]	DRV_SEL3	15	12
SPI0_MO	spi0_conf[3:0]	DRV_SEL3	15	12
SDA1	i2c1_conf[3:0]	DRV_SEL4	3	0

Name	Driving control	Register name	MSB	LSB
SCL1	i2c1_conf[3:0]	DRV_SEL4	3	0
WB_RSTB	fm_conf[3:0]	DRV_SEL4	11	8
GPIO61	fm_conf[3:0]	DRV_SEL4	11	8
GPIO62	fm_conf[3:0]	DRV_SEL4	11	8
WB_SCLK	wbspi_conf[3:0]	DRV_SEL4	15	12
WB_SDATA	wbspi_conf[3:0]	DRV_SEL4	15	12
WB_SEN	wbspi_conf[3:0]	DRV_SEL4	15	12
WB_CTRL0	wb_conf[3:0]	DRV_SEL5	3	0
WB_CTRL1	wb_conf[3:0]	DRV_SEL5	3	0
WB_CTRL2	wb_conf[3:0]	DRV_SEL5	3	0
WB_CTRL3	wb_conf[3:0]	DRV_SEL5	3	0
WB_CTRL4	wb_conf[3:0]	DRV_SEL5	3	0
WB_CTRL5	wb_conf[3:0]	DRV_SEL5	3	0
I2S0_DATA_IN	i2s0_conf[3:0]	DRV_SEL3	7	4
I2S0_LRCK	i2s0_conf[3:0]	DRV_SEL3	7	4
I2S0_BCK	i2s0_conf[3:0]	DRV_SEL3	7	4
SDA0	i2c0_conf[3:0]	DRV_SEL5	7	4
SCL0	i2c0_conf[3:0]	DRV_SEL5	7	4
SDA2	i2c2_conf[3:0]	DRV_SEL5	11	8
SCL2	i2c2_conf[3:0]	DRV_SEL5	11	8
URXD0	uart0_conf[3:0]	DRV_SEL5	15	12
UTXD0	uart0_conf[3:0]	DRV_SEL5	15	12
URXD1	uart0_conf[3:0]	DRV_SEL5	15	12
UTXD1	uart0_conf[3:0]	DRV_SEL5	15	12
LCM_RST	wb_conf[3:0]	DRV_SEL5	3	0
DSI_TE	wb_conf[3:0]	DRV_SEL5	3	0
SPI2_CSN	spi2_conf[3:0]	DRV_SEL6	3	0
SPI2_MI	spi2_conf[3:0]	DRV_SEL6	3	0
SPI2_MO	spi2_conf[3:0]	DRV_SEL6	3	0
SPI2_CK	spi2_conf[3:0]	DRV_SEL6	3	0
CEC	hdmi_conf[3:0]	DRV_SEL6	7	4
HTPLG	hdmi_conf[3:0]	DRV_SEL6	7	4
HDMISCK	hdmi_conf[3:0]	DRV_SEL6	7	4
HDMISD	hdmi_conf[3:0]	DRV_SEL6	7	4
I2S0_MCLK	i2s0_conf[3:0]	DRV_SEL3	7	4
SPI1_CK	spi1_conf[3:0]	DRV_SEL0	7	4
SPDIF_OUT	spdif_conf[3:0]	DRV_SEL8	3	0
SPDIF_IN0	spdif_conf[3:1]	DRV_SEL8	3	0
SPDIF_IN1	spdif_conf[3:2]	DRV_SEL8	3	0
PWM0	pwm_conf[3:0]	DRV_SEL8	7	4
PWM1	pwm_conf[3:0]	DRV_SEL8	7	4

Name	Driving control	Register name	MSB	LSB
PWM2	pwm_conf[3:0]	DRV_SEL8	7	4
PWM3	pwm_conf[3:0]	DRV_SEL8	7	4
PWM4	pwm_conf[3:0]	DRV_SEL8	7	4
AUD_EXT_CK1	audck_conf[3:0]	DRV_SEL8	11	8
AUD_EXT_CK2	audck_conf[3:0]	DRV_SEL8	11	8
SFLASH_IO_3	ext_conf[3:0]	DRV_SEL9	7	4
SFLASH_IO_2	ext_conf[3:0]	DRV_SEL9	7	4
SFLASH_IO_1	ext_conf[3:0]	DRV_SEL9	7	4
SFLASH_IO_0	ext_conf[3:0]	DRV_SEL9	7	4
SFLASH_CS_L	ext_conf[3:0]	DRV_SEL9	7	4
SFLASH_CLK	ext_conf[3:0]	DRV_SEL9	7	4
URTS2	ud_conf[3:0]	DRV_SEL9	11	8
UCTS2	ud_conf[3:0]	DRV_SEL9	11	8
GPIO244	mhl_conf[3:0]	DRV_SEL9	15	12
GPIO245	mhl_conf[3:0]	DRV_SEL9	15	12
MHL_SENCE	mhl_conf[3:0]	DRV_SEL9	15	12
GPIO247	mhl_conf[3:0]	DRV_SEL9	15	12
GPIO248	hdmiout_conf[3:0]	DRV_SELA	3	0
GE2_TXEN	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_TXD3	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_TXD2	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_TXD1	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_TXD0	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_TXCLK	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXCLK	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXD0	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXD1	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXD2	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXD3	rgmii_conf[3:0]	DRV_SELA	11	8
GE2_RXDV	rgmii_conf[3:0]	DRV_SELA	11	8
MDC	rgmii_conf[3:0]	DRV_SELA	11	8
MDIO	rgmii_conf[3:0]	DRV_SELA	11	8
JTAG_RESET	jtag_conf[3:0]	DRV_SEL2	11	8
MSDC1_CMD	0x10005D40	MSDC1_CTRL1	3	0
MSDC1_CLK	0x10005D30	MSDC1_CTRL0	3	0
MSDC1_DAT0	0x10005D50	MSDC1_CTRL2	3	0
MSDC1_DAT1	0x10005D50	MSDC1_CTRL2	3	0
MSDC1_DAT2	0x10005D50	MSDC1_CTRL2	3	0
MSDC1_DAT3	0x10005D50	MSDC1_CTRL2	3	0
MSDC0_DAT7	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_DAT6	0x10005CE0	MSDC0_CTRL2	3	0

Name	Driving control	Register name	MSB	LSB
MSDC0_DAT5	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_DAT4	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_RSTB	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_CMD	0x10005CD0	MSDC0_CTRL1	3	0
MSDC0_CLK	0x10005CC0	MSDC0_CTRL0	3	0
MSDC0_DAT3	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_DAT2	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_DAT1	0x10005CE0	MSDC0_CTRL2	3	0
MSDC0_DAT0	0x10005CE0	MSDC0_CTRL2	3	0
GPIO250	0x10005FC0	MSDC3_CTRL2	3	0
GPIO251	0x10005FC0	MSDC3_CTRL2	3	0
GPIO252	0x10005FC0	MSDC3_CTRL2	3	0
GPIO253	0x10005FC0	MSDC3_CTRL2	3	0
GPIO254	0x10005FC0	MSDC3_CTRL2	3	0
GPIO255	0x10005FC0	MSDC3_CTRL2	3	0
GPIO256	0x10005FC0	MSDC3_CTRL2	3	0
GPIO257	0x10005FC0	MSDC3_CTRL2	3	0

1.4 External Interrupt List

GPIO module provides the external interrupt path for external interrupt controller. When using GPIO as external interrupt source, the GPIO need to be programmed to be GPIO mode (aux. function 0) and input direction.
Most of GPIOs can be used external source. The detail GPIO and EINT source mapping is listed in below table.

Table 1-7: External Interrupt List

Pin NamePAD	GPIO Name	EINT Controller Source
PWRAP_SPI0_MI	GPIO0	148
PWRAP_SPI0_MO	GPIO1	149
PWRAP_INT	GPIO2	150
PWRAP_SPI0_CK	GPIO3	151
PWRAP_SPI0_CSN	GPIO4	152
PWRAP_SPI0_CK2	GPIO5	153
PWRAP_SPI0_CSN2	GPIO6	154
SPI1_CSN	GPIO7	155
SPI1_MI	GPIO8	156
SPI1_MO	GPIO9	157
RTC32K_CK	GPIO10	158
WATCHDOG	GPIO11	159

Pin NamePAD	GPIO Name	EINT Controller Source
SRCLKENA	GPIO12	160
SRCLKENAI	GPIO13	161
URXD2	GPIO14	162
UTXD2	GPIO15	163
PCM_CLK	GPIO18	166
PCM_SYNC	GPIO19	167
EINT0	GPIO22	0
EINT1	GPIO23	1
EINT2	GPIO24	2
EINT3	GPIO25	3
EINT4	GPIO26	4
EINT5	GPIO27	5
EINT6	GPIO28	6
EINT7	GPIO29	7
Iddig	Internal signal	10
USB20_VBUSVALID	Internal signal	11
I2S1_DATA	GPIO33	15
I2S1_DATA_IN	GPIO34	16
I2S1_BCK	GPIO35	17
I2S1_LRCK	GPIO36	18
I2S1_MCLK	GPIO37	19
JTMS	GPIO39	21
JTCK	GPIO40	22
JTDI	GPIO41	23
JTDO	GPIO42	24
NCLE	GPIO43	25
NCEB1	GPIO44	26
NCEB0	GPIO45	27
IR	GPIO46	28
NREB	GPIO47	29
NRNB	GPIO48	30
I2S0_DATA	GPIO49	31
SPI0_CSN	GPIO53	35
SPI0_CK	GPIO54	36
SPI0_MI	GPIO55	37
SPI0_MO	GPIO56	38
SDA1	GPIO57	39
SCL1	GPIO58	40
WB_RSTB	GPIO60	41
GPIO61	GPIO61	42
GPIO62	GPIO62	43

Pin NamePAD	GPIO Name	EINT Controller Source
WB_SCLK	GPIO63	44
WB_SDATA	GPIO64	45
WB_SEN	GPIO65	46
WB_CTRL0	GPIO66	47
WB_CTRL1	GPIO67	48
WB_CTRL2	GPIO68	49
WB_CTRL3	GPIO69	50
WB_CTRL4	GPIO70	51
WB_CTRL5	GPIO71	52
I2S0_DATA_IN	GPIO72	53
I2S0_LRCK	GPIO73	54
I2S0_BCK	GPIO74	55
SDA0	GPIO75	56
SCL0	GPIO76	57
SDA2	GPIO77	58
SCL2	GPIO78	59
URXD0	GPIO79	60
UTXD0	GPIO80	61
URXD1	GPIO81	62
UTXD1	GPIO82	63
LCM_RST	GPIO83	64
DSI_TE	GPIO84	65
GPIO247	GPIO247	69/70
SPI2_CSN	GPIO101	74
SPI2_MI	GPIO102	75
SPI2_MO	GPIO103	76
SPI2_CK	GPIO104	77
MSDC1_CMD	GPIO105	78
MSDC1_CLK	GPIO106	79
MSDC1_DAT0	GPIO107	80
MSDC1_DAT1	GPIO108	81
MSDC1_DAT2	GPIO109	82
MSDC1_DAT3	GPIO110	83
MSDC0_DAT7	GPIO111	84
MSDC0_DAT6	GPIO112	85
MSDC0_DAT5	GPIO113	86
MSDC0_DAT4	GPIO114	87
MSDC0_RSTB	GPIO115	88
MSDC0_CMD	GPIO116	89
MSDC0_CLK	GPIO117	90
MSDC0_DAT3	GPIO118	91

Pin NamePAD	GPIO Name	EINT Controller Source
MSDC0_DAT2	GPIO119	92
MSDC0_DAT1	GPIO120	93
MSDC0_DAT0	GPIO121	94
CEC	GPIO122	95
HTPLG	GPIO123	96
HDMISCK	GPIO124	97
HDMISD	GPIO125	98
I2S0_MCLK	GPIO126	99
SPI1_CK	GPIO199	111
SPDIF_OUT	GPIO200	112
SPDIF_IN0	GPIO201	113
SPDIF_IN1	GPIO202	114
PWM0	GPIO203	115
PWM1	GPIO204	116
PWM2	GPIO205	117
PWM3	GPIO206	118
PWM4	GPIO207	119
AUD_EXT_CK1	GPIO208	120
AUD_EXT_CK2	GPIO209	121
SFLASH_IO_3	GPIO236	122
SFLASH_IO_2	GPIO237	123
SFLASH_IO_1	GPIO238	124
SFLASH_IO_0	GPIO239	125
SFLASH_CS_L	GPIO240	126
SFLASH_CLK	GPIO241	127
URTS2	GPIO242	128
UCTS2	GPIO243	129
GPIO244	GPIO244	130
GPIO245	GPIO245	131
MHL_SENCE	GPIO246	132
GPIO248	GPIO248	133
GPIO250	GPIO250	135
GPIO251	GPIO251	136
GPIO252	GPIO252	137
GPIO253	GPIO253	138
GPIO254	GPIO254	139
GPIO255	GPIO255	140
GPIO256	GPIO256	141
GPIO257	GPIO257	142
JTAG_RESET	GPIO278	147

The debounce feature should be enable when below GPIOs is used as external interrupt source,

Table 1-8: EINT source usage notes

Pin NamePAD	EINT Controller Source (eirq_bus[168:0])	Notes
EINT0	0	Need Enable Debounce Feature
EINT1	1	Need Enable Debounce Feature
EINT2	2	Need Enable Debounce Feature
EINT3	3	Need Enable Debounce Feature
EINT4	4	Need Enable Debounce Feature
EINT5	5	Need Enable Debounce Feature
EINT6	6	Need Enable Debounce Feature
EINT7	7	Need Enable Debounce Feature

1.5 GPIO Usage Tips

MIPI Tx pins in below table are analog and digital shared pins. In GPIO mode, these pins can only be used as GPI. These pins must be set to MIPI mode or GPI mode at the same time to get good performance.

Table 1-9: GPIO use MIPI-DSI Tx pin

MIPI_TCN	GPI95
MIPI_TCP	GPI96
MIPI_TDN1	GPI97
MIPI_TDP1	GPI98
MIPI_TDN0	GPI99
MIPI_TDP0	GPI100

There are two special pins (**I2S0_MCLK** and **I2S1_MCLK**) cannot be used as normal GPIO. These pins will output MCLK after boot-up until software modify their mode selection setting. Because these pins' aux function default are MCLK output

Table 1-10: GPIO use I2S MCLK

I2S0_MCLK	GPIO126
I2S1_MCLK	GPIO37

1.6 Register Definition

Module name: GPIO Base address: (+10005000h)

Address	Name	Width	Register Function
10005000	GPIO_DIR1	16	GPIO Direction Control Register 1
10005010	GPIO_DIR2	16	GPIO Direction Control Register 2
10005020	GPIO_DIR3	16	GPIO Direction Control Register 3
10005030	GPIO_DIR4	16	GPIO Direction Control Register 4
10005040	GPIO_DIR5	16	GPIO Direction Control Register 5
10005050	GPIO_DIR6	16	GPIO Direction Control Register 6
10005060	GPIO_DIR7	16	GPIO Direction Control Register 7
10005070	GPIO_DIR8	16	GPIO Direction Control Register 8
10005080	GPIO_DIR9	16	GPIO Direction Control Register 9
10005090	GPIO_DIR10	16	GPIO Direction Control Register 10
100050A0	GPIO_DIR11	16	GPIO Direction Control Register 11
100050B0	MSDC1_CTRL6	16	MSDC 1 INS Pad Control Register 3
100050C0	GPIO_DIR12	16	GPIO Direction Control Register 12
100050D0	GPIO_DIR13	16	GPIO Direction Control Register 13
100050E0	GPIO_DIR14	16	GPIO Direction Control Register 14
100050F0	GPIO_DIR15	16	GPIO Direction Control Register 15
10005100	GPIO_DIR16	16	GPIO Direction Control Register 16
10005110	GPIO_DIR17	16	GPIO Direction Control Register 17
10005120	GPIO_DIR18	16	GPIO Direction Control Register 18
10005130	SDIO_CTRL4	16	SDIO DATA Pad Control Register 2
10005140	SDIO_CTRL5	16	SDIO DATA Pad Control Register 3
10005150	GPIO_PULLEN1	16	GPIO Pull-up/Pull-down Enable Register 1
10005160	GPIO_PULLEN2	16	GPIO Pull-up/Pull-down Enable Register 2
10005170	GPIO_PULLEN3	16	GPIO Pull-up/Pull-down Enable Register 3
10005180	GPIO_PULLEN4	16	GPIO Pull-up/Pull-down Enable Register 4
10005190	GPIO_PULLEN5	16	GPIO Pull-up/Pull-down Enable Register 5
100051A0	GPIO_PULLEN6	16	GPIO Pull-up/Pull-down Enable Register 6
100051B0	GPIO_PULLEN7	16	GPIO Pull-up/Pull-down Enable Register 7
100051C0	GPIO_PULLEN8	16	GPIO Pull-up/Pull-down Enable Register 8
100051D0	GPIO_PULLEN9	16	GPIO Pull-up/Pull-down Enable Register 9
100051E0	GPIO_PULLEN10	16	GPIO Pull-up/Pull-down Enable Register 10
100051F0	GPIO_PULLEN11	16	GPIO Pull-up/Pull-down Enable Register 11
10005200	GPIO_PULLEN12	16	GPIO Pull-up/Pull-down Enable Register 12
10005210	GPIO_PULLEN13	16	GPIO Pull-up/Pull-down Enable Register 13
10005220	GPIO_PULLEN14	16	GPIO Pull-up/Pull-down Enable Register 14
10005230	GPIO_PULLEN15	16	GPIO Pull-up/Pull-down Enable Register 15

Module name: GPIO Base address: (+10005000h)

10005240	GPIO_PULLEN16	16	GPIO Pull-up/Pull-down Enable Register 16
10005250	GPIO_PULLEN17	16	GPIO Pull-up/Pull-down Enable Register 17
10005260	GPIO_PULLEN18	16	GPIO Pull-up/Pull-down Enable Register 18
10005280	GPIO_PULLSEL1	16	GPIO Pull-up/Pull-down Selection Register 1
10005290	GPIO_PULLSEL2	16	GPIO Pull-up/Pull-down Selection Register 2
100052A0	GPIO_PULLSEL3	16	GPIO Pull-up/Pull-down Selection Register 3
100052B0	GPIO_PULLSEL4	16	GPIO Pull-up/Pull-down Selection Register 4
100052C0	GPIO_PULLSEL5	16	GPIO Pull-up/Pull-down Selection Register 5
100052D0	GPIO_PULLSEL6	16	GPIO Pull-up/Pull-down Selection Register 6
100052E0	GPIO_PULLSEL7	16	GPIO Pull-up/Pull-down Selection Register 7
100052F0	GPIO_PULLSEL8	16	GPIO Pull-up/Pull-down Selection Register 8
10005300	GPIO_PULLSEL9	16	GPIO Pull-up/Pull-down Selection Register 9
10005310	GPIO_PULLSEL10	16	GPIO Pull-up/Pull-down Selection Register 10
10005320	GPIO_PULLSEL11	16	GPIO Pull-up/Pull-down Selection Register 11
10005330	GPIO_PULLSEL12	16	GPIO Pull-up/Pull-down Selection Register 12
10005340	GPIO_PULLSEL13	16	GPIO Pull-up/Pull-down Selection Register 13
10005350	GPIO_PULLSEL14	16	GPIO Pull-up/Pull-down Selection Register 14
10005360	GPIO_PULLSEL15	16	GPIO Pull-up/Pull-down Selection Register 15
10005370	GPIO_PULLSEL16	16	GPIO Pull-up/Pull-down Selection Register 16
10005380	GPIO_PULLSEL17	16	GPIO Pull-up/Pull-down Selection Register 17
10005390	GPIO_PULLSEL18	16	GPIO Pull-up/Pull-down Selection Register 18
100053A0	SDIO_CTRL7	16	SDIO RCLK Pad Control Register 3
10005410	BIAS_CTRL3	16	18OD33 IO Group BIAS Control Register 3
10005420	BIAS_CTRL4	16	18OD33 IO Group BIAS Control Register 4
10005430	SDIO_CTRL8	16	SDIO RCLK SEL Control Register
10005440	OD33_CTRL11	16	18OD33 IO Group TDSEL/RDSEL Control Register 11
10005450	OD33_CTRL12	16	18OD33 IO Group TDSEL/RDSEL Control Register 12
10005460	OD33_CTRL13	16	18OD33 IO Group TDSEL/RDSEL Control Register 13
10005470	OD33_CTRL14	16	18OD33 IO Group TDSEL/RDSEL Control Register 14
100054C0	OD33_CTRL8	16	18OD33 IO Group TDSEL/RDSEL Control Register 8
100054D0	OD33_CTRL9	16	18OD33 IO Group TDSEL/RDSEL Control Register 9
100054E0	OD33_CTRL10	16	18OD33 IO Group TDSEL/RDSEL Control Register 10
10005500	GPIO_DOUT1	16	GPIO Data Output Register 1
10005510	GPIO_DOUT2	16	GPIO Data Output Register 2
10005520	GPIO_DOUT3	16	GPIO Data Output Register 3
10005530	GPIO_DOUT4	16	GPIO Data Output Register 4
10005540	GPIO_DOUT5	16	GPIO Data Output Register 5
10005550	GPIO_DOUT6	16	GPIO Data Output Register 6
10005560	GPIO_DOUT7	16	GPIO Data Output Register 7
10005570	GPIO_DOUT8	16	GPIO Data Output Register 8
10005580	GPIO_DOUT9	16	GPIO Data Output Register 9
10005590	GPIO_DOUT10	16	GPIO Data Output Register 10
100055A0	GPIO_DOUT11	16	GPIO Data Output Register 11
100055B0	GPIO_DOUT12	16	GPIO Data Output Register 12

Module name: GPIO Base address: (+10005000h)

100055C0	GPIO_DOUT13	16	GPIO Data Output Register 13
100055D0	GPIO_DOUT14	16	GPIO Data Output Register 14
100055E0	GPIO_DOUT15	16	GPIO Data Output Register 15
100055F0	GPIO_DOUT16	16	GPIO Data Output Register 16
10005600	GPIO_DOUT17	16	GPIO Data Output Register 17
10005610	GPIO_DOUT18	16	GPIO Data Output Register 18
10005620	SDIO_CTRL6	16	SDIO Pad Control Register
10005630	GPIO_DIN1	16	GPIO Data Input Register 1
10005640	GPIO_DIN2	16	GPIO Data Input Register 2
10005650	GPIO_DIN3	16	GPIO Data Input Register 3
10005660	GPIO_DIN4	16	GPIO Data Input Register 4
10005670	GPIO_DIN5	16	GPIO Data Input Register 5
10005680	GPIO_DIN6	16	GPIO Data Input Register 6
10005690	GPIO_DIN7	16	GPIO Data Input Register 7
100056A0	GPIO_DIN8	16	GPIO Data Input Register 8
100056B0	GPIO_DIN9	16	GPIO Data Input Register 9
100056C0	GPIO_DIN10	16	GPIO Data Input Register 10
100056D0	GPIO_DIN11	16	GPIO Data Input Register 11
100056E0	GPIO_DIN12	16	GPIO Data Input Register 12
100056F0	GPIO_DIN13	16	GPIO Data Input Register 13
10005700	GPIO_DIN14	16	GPIO Data Input Register 14
10005710	GPIO_DIN15	16	GPIO Data Input Register 15
10005720	GPIO_DIN16	16	GPIO Data Input Register 16
10005730	GPIO_DIN17	16	GPIO Data Input Register 17
10005740	GPIO_DIN18	16	GPIO Data Input Register 18
10005760	GPIO_MODE1	16	GPIO Mode Control Register 1
10005770	GPIO_MODE2	16	GPIO Mode Control Register 2
10005780	GPIO_MODE3	16	GPIO Mode Control Register 3
10005790	GPIO_MODE4	16	GPIO Mode Control Register 4
100057A0	GPIO_MODE5	16	GPIO Mode Control Register 5
100057B0	GPIO_MODE6	16	GPIO Mode Control Register 6
100057C0	GPIO_MODE7	16	GPIO Mode Control Register 7
100057D0	GPIO_MODE8	16	GPIO Mode Control Register 8
100057E0	GPIO_MODE9	16	GPIO Mode Control Register 9
100057F0	GPIO_MODE10	16	GPIO Mode Control Register 10
10005800	GPIO_MODE11	16	GPIO Mode Control Register 11
10005810	GPIO_MODE12	16	GPIO Mode Control Register 12
10005820	GPIO_MODE13	16	GPIO Mode Control Register 13
10005830	GPIO_MODE14	16	GPIO Mode Control Register 14
10005840	GPIO_MODE15	16	GPIO Mode Control Register 15
10005850	GPIO_MODE16	16	GPIO Mode Control Register 16
10005860	GPIO_MODE17	16	GPIO Mode Control Register 17
10005870	GPIO_MODE18	16	GPIO Mode Control Register 18
10005880	GPIO_MODE19	16	GPIO Mode Control Register 19

Module name: GPIO Base address: (+10005000h)

10005890	GPIO_MODE20	16	GPIO Mode Control Register 20
100058A0	GPIO_MODE21	16	GPIO Mode Control Register 21
100058B0	GPIO_MODE22	16	GPIO Mode Control Register 22
100058C0	GPIO_MODE23	16	GPIO Mode Control Register 23
100058D0	GPIO_MODE24	16	GPIO Mode Control Register 24
100058E0	GPIO_MODE25	16	GPIO Mode Control Register 25
100058F0	GPIO_MODE26	16	GPIO Mode Control Register 26
10005900	GPIO_MODE27	16	GPIO Mode Control Register 27
10005910	GPIO_MODE28	16	GPIO Mode Control Register 28
10005920	GPIO_MODE29	16	GPIO Mode Control Register 29
10005930	GPIO_MODE30	16	GPIO Mode Control Register 30
10005940	GPIO_MODE31	16	GPIO Mode Control Register 31
10005950	GPIO_MODE32	16	GPIO Mode Control Register 32
10005960	GPIO_MODE33	16	GPIO Mode Control Register 33
10005970	GPIO_MODE34	16	GPIO Mode Control Register 34
10005980	GPIO_MODE35	16	GPIO Mode Control Register 35
10005990	GPIO_MODE36	16	GPIO Mode Control Register 36
100059A0	GPIO_MODE37	16	GPIO Mode Control Register 37
100059B0	GPIO_MODE38	16	GPIO Mode Control Register 38
100059C0	GPIO_MODE39	16	GPIO Mode Control Register 39
100059D0	GPIO_MODE40	16	GPIO Mode Control Register 40
100059E0	GPIO_MODE41	16	GPIO Mode Control Register 41
100059F0	GPIO_MODE42	16	GPIO Mode Control Register 42
10005A00	GPIO_MODE43	16	GPIO Mode Control Register 43
10005A10	GPIO_MODE44	16	GPIO Mode Control Register 44
10005A20	GPIO_MODE45	16	GPIO Mode Control Register 45
10005A30	GPIO_MODE46	16	GPIO Mode Control Register 46
10005A40	GPIO_MODE47	16	GPIO Mode Control Register 47
10005A50	GPIO_MODE48	16	GPIO Mode Control Register 48
10005A60	GPIO_MODE49	16	GPIO Mode Control Register 49
10005A70	GPIO_MODE50	16	GPIO Mode Control Register 50
10005A80	GPIO_MODE51	16	GPIO Mode Control Register 51
10005A90	GPIO_MODE52	16	GPIO Mode Control Register 52
10005AA0	GPIO_MODE53	16	GPIO Mode Control Register 53
10005AB0	GPIO_MODE54	16	GPIO Mode Control Register 54
10005AC0	GPIO_MODE55	16	GPIO Mode Control Register 55
10005AD0	GPIO_MODE56	16	GPIO Mode Control Register 56
10005B10	GPIO_BANK	16	GPIO Misc Control Register
10005B20	IES_EN0	16	GPIO IES Control Register 0
10005B30	IES_EN1	16	GPIO IES Control Register 1
10005B40	IES_EN2	16	GPIO IES Control Register 2
10005B50	SMT_EN0	16	GPIO SMT Control Register 0
10005B60	SMT_EN1	16	GPIO SMT Control Register 1
10005B70	SMT_EN2	16	GPIO SMT Control Register 2

Module name: GPIO Base address: (+10005000h)

10005B80	TDSEL0	16	GPIO TDSEL Control Register 0
10005B90	TDSEL1	16	GPIO TDSEL Control Register 1
10005BA0	TDSEL2	16	GPIO TDSEL Control Register 2
10005BB0	TDSEL3	16	GPIO TDSEL Control Register 3
10005BC0	TDSEL4	16	GPIO TDSEL Control Register 4
10005BD0	TDSEL5	16	GPIO TDSEL Control Register 5
10005BE0	OD33_CTRL4	16	18OD33 IO Group TDSEL/RDSEL Control Register 4
10005BF0	OD33_CTRL5	16	18OD33 IO Group TDSEL/RDSEL Control Register 5
10005C00	OD33_CTRL6	16	18OD33 IO Group TDSEL/RDSEL Control Register 6
10005C10	OD33_CTRL7	16	18OD33 IO Group TDSEL/RDSEL Control Register 7
10005C20	RDSEL0	16	GPIO RDSEL Control Register 0
10005C30	RDSEL1	16	GPIO RDSEL Control Register 1
10005C40	RDSEL2	16	GPIO RDSEL Control Register 2
10005C50	RDSEL3	16	GPIO RDSEL Control Register 3
10005C60	RDSEL4	16	GPIO RDSEL Control Register 4
10005C70	RDSEL5	16	GPIO RDSEL Control Register5
10005C80	DRVN0_EN	16	GPIO Control DDR Register
10005CA0	DRVP0_EN	16	GPIO Control DDR Register
10005CC0	MSDC0_CTRL0	16	MSDC 0 CLK Pad Control Register
10005CD0	MSDC0_CTRL1	16	MSDC 0 CMD Pad Control Register
10005CE0	MSDC0_CTRL2	16	MSDC 0 DATA Pad Control Register 0
10005CF0	MSDC0_CTRL3	16	MSDC 0 DATA Pad Control Register 1
10005D00	MSDC0_CTRL4	16	MSDC 0 DATA Pad Control Register 2
10005D10	MSDC0_CTRL5	16	MSDC 0 DATA Pad Control Register 3
10005D20	MSDC0_CTRL6	16	MSDC 0 Pad Control Register
10005D30	MSDC1_CTRL0	16	MSDC 1 CLK Pad Control Register
10005D40	MSDC1_CTRL1	16	MSDC 1 CMD Pad Control Register
10005D50	MSDC1_CTRL2	16	MSDC 1 DATA Pad Control Register 0
10005D60	MSDC1_CTRL3	16	MSDC 1 DATA Pad Control Register 1
10005D70	MSDC1_CTRL4	16	MSDC 1 DATA Pad Control Register 2
10005D80	MSDC1_CTRL5	16	MSDC 1 Pad Control Register
10005DF0	GPIO_TM	16	GPIO DIR Status Selection Register
10005E00	GPIO_USB	16	USB IDDIG GPIO PULLUP Control Register
10005E10	OD33_CTRL0	16	18OD33 IO Group TDSEL/RDSEL Control Register 0
10005E20	OD33_CTRL1	16	18OD33 IO Group TDSEL/RDSEL Control Register 1
10005E30	OD33_CTRL2	16	18OD33 IO Group TDSEL/RDSEL Control Register 2
10005E40	OD33_CTRL3	16	18OD33 IO Group TDSEL/RDSEL Control Register 3
10005E50	KPAD_CTRL0	16	Keypad ROW Pad R0/R1/PUPD Control Register 0
10005E60	KPAD_CTRL1	16	Keypad COL Pad R0/R1/PUPD Control Register 1
10005E70	EINT_CTRL0	16	EINT Pad R0/R1/PUPD Control Register 0
10005E80	EINT_CTRL1	16	EINT Pad R0/R1/PUPD Control Register 1
10005EB0	BIAS_CTRL0	16	18OD33 IO Group BIAS Control Register 0
10005EC0	BIAS_CTRL1	16	18OD33 IO Group BIAS Control Register 1
10005ED0	BIAS_CTRL2	16	18OD33 IO Group BIAS Control Register 2

Module name: GPIO Base address: (+10005000h)

10005F00	DRV_SEL10	16	GPIO Driving Control Register 10
10005F10	DRV_SEL11	16	GPIO Driving Control Register 11
10005F30	DRV_SEL12	16	GPIO Driving Control Register 12
10005F40	SDIO_CTRL3	16	SDIO DATA Pad Control Register 1
10005F50	DRV_SEL0	16	GPIO Driving Control Register 0
10005F60	DRV_SEL1	16	GPIO Driving Control Register 1
10005F70	DRV_SEL2	16	GPIO Driving Control Register 2
10005F80	DRV_SEL3	16	GPIO Driving Control Register 3
10005F90	DRV_SEL4	16	GPIO Driving Control Register 4
10005FA0	DRV_SEL5	16	GPIO Driving Control Register 5
10005FB0	DRV_SEL6	16	GPIO Driving Control Register 6
10005FC0	SDIO_CTRL2	16	SDIO DATA Pad Control Register 0
10005FD0	DRV_SEL8	16	GPIO Driving Control Register 8
10005FE0	DRV_SEL7	16	GPIO Driving Control Register 7
10005FF0	DRV_SEL9	16	GPIO Driving Control Register 9

10005000 **GPIO_DIR1** **GPIO Direction Control Register 1** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO15_D	GPIO14_D	GPIO13_D	GPIO12_D	GPIO11_D	GPIO10_D	GPIO9_D	GPIO8_D	GPIO7_D	GPIO6_D	GPIO5_D	GPIO4_D	GPIO3_D	GPIO2_D	GPIO1_D	GPIO0_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	GPIO15_D	GPIO15_DIR	**GPIO 15 Direction Control** 0: Input 1: Output
14	GPIO14_D	GPIO14_DIR	**GPIO 14 Direction Control** 0: Input 1: Output
13	GPIO13_D	GPIO13_DIR	**GPIO 13 Direction Control** 0: Input 1: Output
12	GPIO12_D	GPIO12_DIR	**GPIO 12 Direction Control** 0: Input 1: Output
11	GPIO11_D	GPIO11_DIR	**GPIO 11 Direction Control** 0: Input 1: Output
10	GPIO10_D	GPIO10_DIR	**GPIO 10 Direction Control** 0: Input 1: Output
9	GPIO9_D	GPIO9_DIR	**GPIO 9 Direction Control**

Bit(s)	Mnemonic	Name	Description
			0: Input 1: Output
8	**GPIO8_D**	GPIO8_DIR	**GPIO 8 Direction Control** 0: Input 1: Output
7	**GPIO7_D**	GPIO7_DIR	**GPIO 7 Direction Control** 0: Input 1: Output
6	**GPIO6_D**	GPIO6_DIR	**GPIO 6 Direction Control** 0: Input 1: Output
5	**GPIO5_D**	GPIO5_DIR	**GPIO 5 Direction Control** 0: Input 1: Output
4	**GPIO4_D**	GPIO4_DIR	**GPIO 4 Direction Control** 0: Input 1: Output
3	**GPIO3_D**	GPIO3_DIR	**GPIO 3 Direction Control** 0: Input 1: Output
2	**GPIO2_D**	GPIO2_DIR	**GPIO 2 Direction Control** 0: Input 1: Output
1	**GPIO1_D**	GPIO1_DIR	**GPIO 1 Direction Control** 0: Input 1: Output
0	**GPIO0_D**	GPIO0_DIR	**GPIO 0 Direction Control** 0: Input 1: Output

10005010 GPIO_DIR2 GPIO Direction Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO31_D	GPIO30_D	GPIO29_D	GPIO28_D	GPIO27_D	GPIO26_D	GPIO25_D	GPIO24_D	GPIO23_D	GPIO22_D	GPIO21_D	GPIO20_D	GPIO19_D	GPIO18_D	GPIO17_D	GPIO16_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO31_D**	GPIO31_DIR	**GPIO 31 Direction Control** 0: Input 1: Output
14	**GPIO30_D**	GPIO30_DIR	**GPIO 30 Direction Control** 0: Input 1: Output
13	**GPIO29_D**	GPIO29_DIR	**GPIO 29 Direction Control** 0: Input 1: Output

Bit(s)	Mnemonic	Name	Description
12	**GPIO28_D**	GPIO28_DIR	**GPIO 28 Direction Control** 0: Input 1: Output
11	**GPIO27_D**	GPIO27_DIR	**GPIO 27 Direction Control** 0: Input 1: Output
10	**GPIO26_D**	GPIO26_DIR	**GPIO 26 Direction Control** 0: Input 1: Output
9	**GPIO25_D**	GPIO25_DIR	**GPIO 25 Direction Control** 0: Input 1: Output
8	**GPIO24_D**	GPIO24_DIR	**GPIO 24 Direction Control** 0: Input 1: Output
7	**GPIO23_D**	GPIO23_DIR	**GPIO 23 Direction Control** 0: Input 1: Output
6	**GPIO22_D**	GPIO22_DIR	**GPIO 22 Direction Control** 0: Input 1: Output
5	**GPIO21_D**	GPIO21_DIR	**GPIO 21 Direction Control** 0: Input 1: Output
4	**GPIO20_D**	GPIO20_DIR	**GPIO 20 Direction Control** 0: Input 1: Output
3	**GPIO19_D**	GPIO19_DIR	**GPIO 19 Direction Control** 0: Input 1: Output
2	**GPIO18_D**	GPIO18_DIR	**GPIO 18 Direction Control** 0: Input 1: Output
1	**GPIO17_D**	GPIO17_DIR	**GPIO 17 Direction Control** 0: Input 1: Output
0	**GPIO16_D**	GPIO16_DIR	**GPIO 16 Direction Control** 0: Input 1: Output

10005020 GPIO_DIR3 GPIO Direction Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO47_D	GPIO46_D	GPIO45_D	GPIO44_D	GPIO43_D	GPIO42_D	GPIO41_D	GPIO40_D	GPIO39_D	GPIO38_D	GPIO37_D	GPIO36_D	GPIO35_D	GPIO34_D	GPIO33_D	GPIO32_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO47_D**	GPIO47_DIR	**GPIO 47 Direction Control** 0: Input 1: Output
14	**GPIO46_D**	GPIO46_DIR	**GPIO 46 Direction Control** 0: Input 1: Output
13	**GPIO45_D**	GPIO45_DIR	**GPIO 45 Direction Control** 0: Input 1: Output
12	**GPIO44_D**	GPIO44_DIR	**GPIO 44 Direction Control** 0: Input 1: Output
11	**GPIO43_D**	GPIO43_DIR	**GPIO 43 Direction Control** 0: Input 1: Output
10	**GPIO42_D**	GPIO42_DIR	**GPIO 42 Direction Control** 0: Input 1: Output
9	**GPIO41_D**	GPIO41_DIR	**GPIO 41 Direction Control** 0: Input 1: Output
8	**GPIO40_D**	GPIO40_DIR	**GPIO 40 Direction Control** 0: Input 1: Output
7	**GPIO39_D**	GPIO39_DIR	**GPIO 39 Direction Control** 0: Input 1: Output
6	**GPIO38_D**	GPIO38_DIR	**GPIO 38 Direction Control** 0: Input 1: Output
5	**GPIO37_D**	GPIO37_DIR	**GPIO 37 Direction Control** 0: Input 1: Output
4	**GPIO36_D**	GPIO36_DIR	**GPIO 36 Direction Control** 0: Input 1: Output
3	**GPIO35_D**	GPIO35_DIR	**GPIO 35 Direction Control** 0: Input 1: Output
2	**GPIO34_D**	GPIO34_DIR	**GPIO 34 Direction Control** 0: Input 1: Output
1	**GPIO33_D**	GPIO33_DIR	**GPIO 33 Direction Control** 0: Input 1: Output
0	**GPIO32_D**	GPIO32_DIR	**GPIO 32 Direction Control** 0: Input 1: Output

10005030 GPIO_DIR4 GPIO Direction Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO63_D	GPIO62_D	GPIO61_D	GPIO60_D	GPIO59_D	GPIO58_D	GPIO57_D	GPIO56_D	GPIO55_D	GPIO54_D	GPIO53_D	GPIO52_D	GPIO51_D	GPIO50_D	GPIO49_D	GPIO48_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO63_D**	GPIO63_DIR	**GPIO 63 Direction Control** 0: Input 1: Output
14	**GPIO62_D**	GPIO62_DIR	**GPIO 62 Direction Control** 0: Input 1: Output
13	**GPIO61_D**	GPIO61_DIR	**GPIO 61 Direction Control** 0: Input 1: Output
12	**GPIO60_D**	GPIO60_DIR	**GPIO 60 Direction Control** 0: Input 1: Output
11	**Reserved**	Reserved	**Reserved**
10	**GPIO58_D**	GPIO58_DIR	**GPIO 58 Direction Control** 0: Input 1: Output
9	**GPIO57_D**	GPIO57_DIR	**GPIO 57 Direction Control** 0: Input 1: Output
8	**GPIO56_D**	GPIO56_DIR	**GPIO 56 Direction Control** 0: Input 1: Output
7	**GPIO55_D**	GPIO55_DIR	**GPIO 55 Direction Control** 0: Input 1: Output
6	**GPIO54_D**	GPIO54_DIR	**GPIO 54 Direction Control** 0: Input 1: Output
5	**GPIO53_D**	GPIO53_DIR	**GPIO 53 Direction Control** 0: Input 1: Output
4	**GPIO52_D**	GPIO52_DIR	**GPIO 52 Direction Control** 0: Input 1: Output
3	**GPIO51_D**	GPIO51_DIR	**GPIO 51 Direction Control** 0: Input 1: Output
2	**GPIO50_D**	GPIO50_DIR	**GPIO 50 Direction Control** 0: Input 1: Output
1	**GPIO49_D**	GPIO49_DIR	**GPIO 49 Direction Control** 0: Input

Bit(s)	Mnemonic	Name	Description
			1: Output
0	GPIO48_D	GPIO48_DIR	**GPIO 48 Direction Control** 0: Input 1: Output

10005040 GPIO_DIR5 GPIO Direction Control Register 5 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO79_D	GPIO78_D	GPIO77_D	GPIO76_D	GPIO75_D	GPIO74_D	GPIO73_D	GPIO72_D	GPIO71_D	GPIO70_D	GPIO69_D	GPIO68_D	GPIO67_D	GPIO66_D	GPIO65_D	GPIO64_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	GPIO79_D	GPIO79_DIR	**GPIO 79 Direction Control** 0: Input 1: Output
14	GPIO78_D	GPIO78_DIR	**GPIO 78 Direction Control** 0: Input 1: Output
13	GPIO77_D	GPIO77_DIR	**GPIO 77 Direction Control** 0: Input 1: Output
12	GPIO76_D	GPIO76_DIR	**GPIO 76 Direction Control** 0: Input 1: Output
11	GPIO75_D	GPIO75_DIR	**GPIO 75 Direction Control** 0: Input 1: Output
10	GPIO74_D	GPIO74_DIR	**GPIO 74 Direction Control** 0: Input 1: Output
9	GPIO73_D	GPIO73_DIR	**GPIO 73 Direction Control** 0: Input 1: Output
8	GPIO72_D	GPIO72_DIR	**GPIO 72 Direction Control** 0: Input 1: Output
7	GPIO71_D	GPIO71_DIR	**GPIO 71 Direction Control** 0: Input 1: Output
6	GPIO70_D	GPIO70_DIR	**GPIO 70 Direction Control** 0: Input 1: Output
5	GPIO69_D	GPIO69_DIR	**GPIO 69 Direction Control** 0: Input 1: Output

Bit(s)	Mnemonic	Name	Description
4	**GPIO68_D**	GPIO68_DIR	**GPIO 68 Direction Control** 0: Input 1: Output
3	**GPIO67_D**	GPIO67_DIR	**GPIO 67 Direction Control** 0: Input 1: Output
2	**GPIO66_D**	GPIO66_DIR	**GPIO 66 Direction Control** 0: Input 1: Output
1	**GPIO65_D**	GPIO65_DIR	**GPIO 65 Direction Control** 0: Input 1: Output
0	**GPIO64_D**	GPIO64_DIR	**GPIO 64 Direction Control** 0: Input 1: Output

10005050 GPIO_DIR6 GPIO Direction Control Register 6 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO95_D	GPIO94_D	GPIO93_D	GPIO92_D	GPIO91_D	GPIO90_D	GPIO89_D	GPIO88_D	GPIO87_D	GPIO86_D	GPIO85_D	GPIO84_D	GPIO83_D	GPIO82_D	GPIO81_D	GPIO80_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO95_D**	GPIO95_DIR	**GPIO 95 Direction Control** 0: Input 1: Output
14:11	**Reserved**	Reserved	**Reserved**
10	**GPIO90_D**	GPIO90_DIR	**GPIO 90 Direction Control** 0: Input 1: Output
9	**GPIO89_D**	GPIO89_DIR	**GPIO 89 Direction Control** 0: Input 1: Output
8	**GPIO88_D**	GPIO88_DIR	**GPIO 88 Direction Control** 0: Input 1: Output
7	**GPIO87_D**	GPIO87_DIR	**GPIO 87 Direction Control** 0: Input 1: Output
6	**GPIO86_D**	GPIO86_DIR	**GPIO 86 Direction Control** 0: Input 1: Output
5	**GPIO85_D**	GPIO85_DIR	**GPIO 85 Direction Control** 0: Input 1: Output

Bit(s)	Mnemonic	Name	Description
4	**GPIO84_D**	GPIO84_DIR	**GPIO 84 Direction Control** 0: Input 1: Output
3	**GPIO83_D**	GPIO83_DIR	**GPIO 83 Direction Control** 0: Input 1: Output
2	**GPIO82_D**	GPIO82_DIR	**GPIO 82 Direction Control** 0: Input 1: Output
1	**GPIO81_D**	GPIO81_DIR	**GPIO 81 Direction Control** 0: Input 1: Output
0	**GPIO80_D**	GPIO80_DIR	**GPIO 80 Direction Control** 0: Input 1: Output

10005060 GPIO_DIR7 GPIO Direction Control Register 7 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO111_D	GPIO110_D	GPIO109_D	GPIO108_D	GPIO107_D	GPIO106_D	GPIO105_D	GPIO104_D	GPIO103_D	GPIO102_D	GPIO101_D	GPIO100_D	GPIO99_D	GPIO98_D	GPIO97_D	GPIO96_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO111_D**	GPIO111_DIR	**GPIO 111 Direction Control** 0: Input 1: Output
14	**GPIO110_D**	GPIO110_DIR	**GPIO 110 Direction Control** 0: Input 1: Output
13	**GPIO109_D**	GPIO109_DIR	**GPIO 109 Direction Control** 0: Input 1: Output
12	**GPIO108_D**	GPIO108_DIR	**GPIO 108 Direction Control** 0: Input 1: Output
11	**GPIO107_D**	GPIO107_DIR	**GPIO 107 Direction Control** 0: Input 1: Output
10	**GPIO106_D**	GPIO106_DIR	**GPIO 106 Direction Control** 0: Input 1: Output
9	**GPIO105_D**	GPIO105_DIR	**GPIO 105 Direction Control** 0: Input 1: Output
8	**GPIO104**	GPIO104_DIR	**GPIO 104 Direction Control**

Bit(s)	Mnemonic	Name	Description
	_D		0: Input 1: Output
7	GPIO103_D	GPIO103_DIR	**GPIO 103 Direction Control** 0: Input 1: Output
6	GPIO102_D	GPIO102_DIR	**GPIO 102 Direction Control** 0: Input 1: Output
5	GPIO101_D	GPIO101_DIR	**GPIO 101 Direction Control** 0: Input 1: Output
4	GPIO100_D	GPIO100_DIR	**GPIO 100 Direction Control** 0: Input 1: Output
3	GPIO99_D	GPIO99_DIR	**GPIO 99 Direction Control** 0: Input 1: Output
2	GPIO98_D	GPIO98_DIR	**GPIO 98 Direction Control** 0: Input 1: Output
1	GPIO97_D	GPIO97_DIR	**GPIO 97 Direction Control** 0: Input 1: Output
0	GPIO96_D	GPIO96_DIR	**GPIO 96 Direction Control** 0: Input 1: Output

10005070 GPIO_DIR8 GPIO Direction Control Register 8 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO127_D	GPIO126_D	GPIO125_D	GPIO124_D	GPIO123_D	GPIO122_D	GPIO121_D	GPIO120_D	GPIO119_D	GPIO118_D	GPIO117_D	GPIO116_D	GPIO115_D	GPIO114_D	GPIO113_D	GPIO112_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	Reserved	Reserved	**Reserved**
14	GPIO126_D	GPIO126_DIR	**GPIO 126 Direction Control** 0: Input 1: Output
13	GPIO125_D	GPIO125_DIR	**GPIO 125 Direction Control** 0: Input 1: Output
12	GPIO124_D	GPIO124_DIR	**GPIO 124 Direction Control** 0: Input 1: Output
11	GPIO123	GPIO123_DIR	**GPIO 123 Direction Control**

Bit(s)	Mnemonic	Name	Description
	_D		0: Input 1: Output
10	GPIO122_D	GPIO122_DIR	**GPIO 122 Direction Control** 0: Input 1: Output
9	GPIO121_D	GPIO121_DIR	**GPIO 121 Direction Control** 0: Input 1: Output
8	GPIO120_D	GPIO120_DIR	**GPIO 120 Direction Control** 0: Input 1: Output
7	GPIO119_D	GPIO119_DIR	**GPIO 119 Direction Control** 0: Input 1: Output
6	GPIO118_D	GPIO118_DIR	**GPIO 118 Direction Control** 0: Input 1: Output
5	GPIO117_D	GPIO117_DIR	**GPIO 117 Direction Control** 0: Input 1: Output
4	GPIO116_D	GPIO116_DIR	**GPIO 116 Direction Control** 0: Input 1: Output
3	GPIO115_D	GPIO115_DIR	**GPIO 115 Direction Control** 0: Input 1: Output
2	GPIO114_D	GPIO114_DIR	**GPIO 114 Direction Control** 0: Input 1: Output
1	GPIO113_D	GPIO113_DIR	**GPIO 113 Direction Control** 0: Input 1: Output
0	GPIO112_D	GPIO112_DIR	**GPIO 112 Direction Control** 0: Input 1: Output

10005080 **GPIO_DIR9** **GPIO Direction Control Register 9** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO143_D	GPIO142_D	GPIO141_D	GPIO140_D	GPIO139_D	GPIO138_D	GPIO137_D	GPIO136_D	GPIO135_D	GPIO134_D	GPIO133_D	GPIO132_D	GPIO131_D	GPIO130_D	GPIO129_D	GPIO128_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005090 GPIO_DIR10 GPIO Direction Control Register 10 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO159_D	GPIO158_D	GPIO157_D	GPIO156_D	GPIO155_D	GPIO154_D	GPIO153_D	GPIO152_D	GPIO151_D	GPIO150_D	GPIO149_D	GPIO148_D	GPIO147_D	GPIO146_D	GPIO145_D	GPIO144_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100050A0 GPIO_DIR11 GPIO Direction Control Register 11 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO175_D	GPIO174_D	GPIO173_D	GPIO172_D	GPIO171_D	GPIO170_D	GPIO169_D	GPIO168_D	GPIO167_D	GPIO166_D	GPIO165_D	GPIO164_D	GPIO163_D	GPIO162_D	GPIO161_D	GPIO160_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100050B0 MSDC1_CTRL6 MSDC 1 INS Pad Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	MS1INS_BACKUP1												MS1INS_SMT	MS1INS_BACKUP0		
Type	RW												RW	RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:4		MS1INS_BACKUP1	Reserved
3		MS1INS_SMT	Schmitter Trigger 0:Disable 1:Enable
2:0		MS1INS_BACKUP0	Reserved

100050C0 **GPIO_DIR12** **GPIO Direction Control Register 12** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO191_D	GPIO190_D	GPIO189_D	GPIO188_D	GPIO187_D	GPIO186_D	GPIO185_D	GPIO184_D	GPIO183_D	GPIO182_D	GPIO181_D	GPIO180_D	GPIO179_D	GPIO178_D	GPIO177_D	GPIO176_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	GPIO191_D	GPIO191_DIR	**GPIO 191 Direction Control** 0: Input 1: Output
14	GPIO190_D	GPIO190_DIR	**GPIO 190 Direction Control** 0: Input 1: Output
13	GPIO189_D	GPIO189_DIR	**GPIO 189 Direction Control** 0: Input 1: Output
12	GPIO188_D	GPIO188_DIR	**GPIO 188 Direction Control** 0: Input 1: Output
11:0	**Reserved**	Reserved	**Reserved**

100050D0 **GPIO_DIR13** **GPIO Direction Control Register 13** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO207_D	GPIO206_D	GPIO205_D	GPIO204_D	GPIO203_D	GPIO202_D	GPIO201_D	GPIO200_D	GPIO199_D	GPIO198_D	GPIO197_D	GPIO196_D	GPIO195_D	GPIO194_D	GPIO193_D	GPIO192_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	GPIO207_D	GPIO207_DIR	**GPIO 207 Direction Control** 0: Input 1: Output
14	GPIO206_D	GPIO206_DIR	**GPIO 206 Direction Control** 0: Input 1: Output
13	GPIO205_D	GPIO205_DIR	**GPIO 205 Direction Control** 0: Input 1: Output
12	GPIO204_D	GPIO204_DIR	**GPIO 204 Direction Control** 0: Input 1: Output

Bit(s)	Mnemonic	Name	Description
11	GPIO203_D	GPIO203_DIR	**GPIO 203 Direction Control** 0: Input 1: Output
10	GPIO202_D	GPIO202_DIR	**GPIO 202 Direction Control** 0: Input 1: Output
9	GPIO201_D	GPIO201_DIR	**GPIO 201 Direction Control** 0: Input 1: Output
8	GPIO200_D	GPIO200_DIR	**GPIO 200 Direction Control** 0: Input 1: Output
7	GPIO199_D	GPIO199_DIR	**GPIO 199 Direction Control** 0: Input 1: Output
6	GPIO198_D	GPIO198_DIR	**GPIO 198 Direction Control** 0: Input 1: Output
5	GPIO197_D	GPIO197_DIR	**GPIO 197 Direction Control** 0: Input 1: Output
4	GPIO196_D	GPIO196_DIR	**GPIO 196 Direction Control** 0: Input 1: Output
3	GPIO195_D	GPIO195_DIR	**GPIO 195 Direction Control** 0: Input 1: Output
2	GPIO194_D	GPIO194_DIR	**GPIO 194 Direction Control** 0: Input 1: Output
1	GPIO193_D	GPIO193_DIR	**GPIO 193 Direction Control** 0: Input 1: Output
0	GPIO192_D	GPIO192_DIR	**GPIO 192 Direction Control** 0: Input 1: Output

100050E0 **GPIO_DIR14** **GPIO Direction Control Register 14** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO23_D	GPIO22_D	GPIO21_D	GPIO20_D	GPIO219_D	GPIO218_D	GPIO217_D	GPIO216_D	GPIO215_D	GPIO214_D	GPIO213_D	GPIO212_D	GPIO211_D	GPIO210_D	GPIO209_D	GPIO208_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15:2	**Reserved**	Reserved	**Reserved**
1	**GPIO209_D**	GPIO209_DIR	**GPIO 209 Direction Control** 0: Input 1: Output
0	**GPIO208_D**	GPIO208_DIR	**GPIO 208 Direction Control** 0: Input 1: Output

100050F0 **GPIO_DIR15** **GPIO Direction Control Register 15** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO239_D	GPIO238_D	GPIO237_D	GPIO236_D	GPIO235_D	GPIO234_D	GPIO233_D	GPIO232_D	GPIO231_D	GPIO230_D	GPIO229_D	GPIO228_D	GPIO227_D	GPIO226_D	GPIO225_D	GPIO224_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO239_D**	GPIO239_DIR	**GPIO 239 Direction Control** 0: Input 1: Output
14	**GPIO238_D**	GPIO238_DIR	**GPIO 238 Direction Control** 0: Input 1: Output
13	**GPIO237_D**	GPIO237_DIR	**GPIO 237 Direction Control** 0: Input 1: Output
12	**GPIO236_D**	GPIO236_DIR	**GPIO 236 Direction Control** 0: Input 1: Output
11:0	**Reserved**	Reserved	**Reserved**

10005100 **GPIO_DIR16** **GPIO Direction Control Register 16** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPIO255_D	GPIO254_D	GPIO253_D	GPIO252_D	GPIO251_D	GPIO250_D	GPIO249_D	GPIO248_D	GPIO247_D	GPIO246_D	GPIO245_D	GPIO244_D	GPIO243_D	GPIO242_D	GPIO241_D	GPIO240_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO255_D**	GPIO255_DIR	**GPIO 255 Direction Control** 0: Input 1: Output

Bit(s)	Mnemonic	Name	Description
14	GPIO254_D	GPIO254_DIR	**GPIO 254 Direction Control** 0: Input 1: Output
13	GPIO253_D	GPIO253_DIR	**GPIO 253 Direction Control** 0: Input 1: Output
12	GPIO252_D	GPIO252_DIR	**GPIO 252 Direction Control** 0: Input 1: Output
11	GPIO251_D	GPIO251_DIR	**GPIO 251 Direction Control** 0: Input 1: Output
10	GPIO250_D	GPIO250_DIR	**GPIO 250 Direction Control** 0: Input 1: Output
9	GPIO249_D	GPIO249_DIR	**GPIO 249 Direction Control** 0: Input 1: Output
8	GPIO248_D	GPIO248_DIR	**GPIO 248 Direction Control** 0: Input 1: Output
7	GPIO247_D	GPIO247_DIR	**GPIO 247 Direction Control** 0: Input 1: Output
6	GPIO246_D	GPIO246_DIR	**GPIO 246 Direction Control** 0: Input 1: Output
5	GPIO245_D	GPIO245_DIR	**GPIO 245 Direction Control** 0: Input 1: Output
4	GPIO244_D	GPIO244_DIR	**GPIO 244 Direction Control** 0: Input 1: Output
3	GPIO243_D	GPIO243_DIR	**GPIO 243 Direction Control** 0: Input 1: Output
2	GPIO242_D	GPIO242_DIR	**GPIO 242 Direction Control** 0: Input 1: Output
1	GPIO241_D	GPIO241_DIR	**GPIO 241 Direction Control** 0: Input 1: Output
0	GPIO240_D	GPIO240_DIR	**GPIO 240 Direction Control** 0: Input 1: Output

10005110 GPIO_DIR17 GPIO Direction Control Register 17 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI

10005110 GPIO_DIR17 GPIO Direction Control Register 17 0000

	O271_D	O270_D	O269_D	O268_D	O267_D	O266_D	O265_D	O264_D	O263_D	O262_D	O261_D	O260_D	O259_D	O258_D	O257_D	O256_D
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO271_D**	GPIO271_DIR	**GPIO 271 Direction Control** 0: Input 1: Output
14	**GPIO270_D**	GPIO270_DIR	**GPIO 270 Direction Control** 0: Input 1: Output
13	**GPIO269_D**	GPIO269_DIR	**GPIO 269 Direction Control** 0: Input 1: Output
12	**GPIO268_D**	GPIO268_DIR	**GPIO 268 Direction Control** 0: Input 1: Output
11	**GPIO267_D**	GPIO267_DIR	**GPIO 267 Direction Control** 0: Input 1: Output
10	**GPIO266_D**	GPIO266_DIR	**GPIO 266 Direction Control** 0: Input 1: Output
9	**GPIO265_D**	GPIO265_DIR	**GPIO 265 Direction Control** 0: Input 1: Output
8	**GPIO264_D**	GPIO264_DIR	**GPIO 264 Direction Control** 0: Input 1: Output
7	**GPIO263_D**	GPIO263_DIR	**GPIO 263 Direction Control** 0: Input 1: Output
6	**GPIO262_D**	GPIO262_DIR	**GPIO 262 Direction Control** 0: Input 1: Output
5	**GPIO261_D**	GPIO261_DIR	**GPIO 261 Direction Control** 0: Input 1: Output
4	**GPIO260_D**	GPIO260_DIR	**GPIO 260 Direction Control** 0: Input 1: Output
3	**GPIO259_D**	GPIO259_DIR	**GPIO 259 Direction Control** 0: Input 1: Output
2	**GPIO258_D**	GPIO258_DIR	**GPIO 258 Direction Control** 0: Input 1: Output
1	**GPIO257_D**	GPIO257_DIR	**GPIO 257 Direction Control** 0: Input

Bit(s)	Mnemonic	Name	Description
			1: Output
0	GPIO256_D	GPIO256_DIR	**GPIO 256 Direction Control** 0: Input 1: Output

10005120 GPIO_DIR18 GPIO Direction Control Register 18 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									GPIO279_D	GPIO278_D	GPIO277_D	GPIO276_D	GPIO275_D	GPIO274_D	GPIO273_D	GPIO272_D
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	Reserved	Reserved	**Reserved**
6	GPIO278_D	GPIO278_DIR	**GPIO 278 Direction Control** 0: Input 1: Output
5	Reserved	Reserved	**Reserved**
4	GPIO276_D	GPIO276_DIR	**GPIO 276 Direction Control** 0: Input 1: Output
3	GPIO275_D	GPIO275_DIR	**GPIO 275 Direction Control** 0: Input 1: Output
2	GPIO274_D	GPIO274_DIR	**GPIO 274 Direction Control** 0: Input 1: Output
1	Reserved	Reserved	**Reserved**
0	GPIO272_D	GPIO272_DIR	**GPIO 272 Direction Control** 0: Input 1: Output

10005130 MSDC3_CTRL4 MSDC 3DATA Pad Control Register 2 4444

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDAT7_SMT	MSDAT7_Ro	MSDAT7_R1	MSDAT7_PUD	MSDAT6_SMT	MSDAT6_Ro	MSDAT6_R1	MSDAT6_PUPD	MSDAT5_SMT	MSDAT5_Ro	MSDAT5_R1	MSDAT5_PUPD	MSDAT4_SMT	MSDAT4_Ro	MSDAT4_R1	MSDAT4_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15		MS3DAT7_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
14		MS3DAT7_R0	**10K resistot control**
13		MS3DAT7_R1	**50K resistor control**
12		MS3DAT7_PUPD	**pull-up(0)/pull-down(1) control**
11		MS3DAT6_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS3DAT6_R0	**10K resistot control**
9		MS3DAT6_R1	**50K resistor control**
8		MS3DAT6_PUPD	**pull-up(0)/pull-down(1) control**
7		MS3DAT5_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
6		MS3DAT5_R0	**10K resistot control**
5		MS3DAT5_R1	**50K resistor control**
4		MS3DAT5_PUPD	**pull-up(0)/pull-down(1) control**
3		MS3DAT4_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
2		MS3DAT4_R0	**10K resistot control**
1		MS3DAT4_R1	**50K resistor control**
0		MS3DAT4_PUPD	**pull-up(0)/pull-down(1) control**

10005140 **MSDC3_CTRL5** **MSDC3 DATA Pad Control Register 3** **0404**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS0DAT_BACKUP					MS1INS_R0	MS1INS_R1	MS1INS_PUPD	MS3DAT_BACKUP0				MS3RSTB_SMT	MS3RSTB_R0	MS3RSTB_R1	MS3RSTB_PUPD
Type	RW					RW	RW	RW	RW				RW	RW	RW	RW
Reset	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15:11		MS0DAT_BACKUP	**Reserved**
10		MS1INS_R0	**10K resistot control**
9		MS1INS_R1	**50K resistor control**
8		MS1INS_PUPD	**pull-up(0)/pull-down(1) control**
7:4		MS3DAT_BACKUP0	**Reserved**
3		MS3RSTB_SMT	**Schmitter Trigger Control**

Bit(s)	Mnemonic	Name	Description
			0:Disable 1:Enable
2		MS3RSTB_R0	**10K resistot control**
1		MS3RSTB_R1	**50K resistor control**
0		MS3RSTB_PUPD	**pull-up(0)/pull-down(1) control**

10005150 **GPIO_PULLEN1** **GPIO Pull-up/Pull-down Enable Register 1** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO15_P	GPIO14_P	GPIO13_P	GPIO12_P	GPIO11_P	GPIO10_P	GPIO9_P	GPIO8_P	GPIO7_P	GPIO6_P	GPIO5_P	GPIO4_P	GPIO3_P	GPIO2_P	GPIO1_P	GPIO0_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO15_P**	GPIO15_PULLEN	**GPIO 15 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO14_P**	GPIO14_PULLEN	**GPIO 14 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO13_P**	GPIO13_PULLEN	**GPIO 13 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO12_P**	GPIO12_PULLEN	**GPIO 12 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO11_P**	GPIO11_PULLEN	**GPIO 11 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO10_P**	GPIO10_PULLEN	**GPIO 10 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO9_P**	GPIO9_PULLEN	**GPIO 9 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO8_P**	GPIO8_PULLEN	**GPIO 8 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO7_P**	GPIO7_PULLEN	**GPIO 7 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO6_P**	GPIO6_PULLEN	**GPIO 6 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO5_P**	GPIO5_PULLEN	**GPIO 5 Pull-up/Pull-down Enable**

Bit(s)	Mnemonic	Name	Description
			0: Disable 1: Enable
4	**GPIO4_P**	GPIO4_PULLEN	**GPIO 4 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO3_P**	GPIO3_PULLEN	**GPIO 3 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO2_P**	GPIO2_PULLEN	**GPIO 2 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO1_P**	GPIO1_PULLEN	**GPIO 1 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO0_P**	GPIO0_PULLEN	**GPIO 0 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005160 **GPIO_PULLEN2** **GPIO Pull-up/Pull-down Enable Register 2** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO31_P	GPIO30_P	GPIO29_P	GPIO28_P	GPIO27_P	GPIO26_P	GPIO25_P	GPIO24_P	GPIO23_P	GPIO22_P	GPIO21_P	GPIO20_P	GPIO19_P	GPIO18_P	GPIO17_P	GPIO16_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO31_P**	GPIO31_PULLEN	**GPIO 31 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO30_P**	GPIO30_PULLEN	**GPIO 30 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO29_P**	GPIO29_PULLEN	**GPIO 29 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO28_P**	GPIO28_PULLEN	**GPIO 28 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO27_P**	GPIO27_PULLEN	**GPIO 27 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO26_P**	GPIO26_PULLEN	**GPIO 26 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO25_P**	GPIO25_PULLEN	**GPIO 25 Pull-up/Pull-down Enable** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
8	**GPIO24_P**	GPIO24_PULLEN	**GPIO 24 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO23_P**	GPIO23_PULLEN	**GPIO 23 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO22_P**	GPIO22_PULLEN	**GPIO 22 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO21_P**	GPIO21_PULLEN	**GPIO 21 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO20_P**	GPIO20_PULLEN	**GPIO 20 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO19_P**	GPIO19_PULLEN	**GPIO 19 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO18_P**	GPIO18_PULLEN	**GPIO 18 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO17_P**	GPIO17_PULLEN	**GPIO 17 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO16_P**	GPIO16_PULLEN	**GPIO 16 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005170 **GPIO_PULLEN3** **GPIO Pull-up/Pull-down Enable Register 3** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO47_P	GPIO46_P	GPIO45_P	GPIO44_P	GPIO43_P	GPIO42_P	GPIO41_P	GPIO40_P	GPIO39_P	GPIO38_P	GPIO37_P	GPIO36_P	GPIO35_P	GPIO34_P	GPIO33_P	GPIO32_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO47_P**	GPIO47_PULLEN	**GPIO 47 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO46_P**	GPIO46_PULLEN	**GPIO 46 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO45_P**	GPIO45_PULLEN	**GPIO 45 Pull-up/Pull-down Enable** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
12	**GPIO44_P**	GPIO44_PULLEN	**GPIO 44 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO43_P**	GPIO43_PULLEN	**GPIO 43 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO42_P**	GPIO42_PULLEN	**GPIO 42 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO41_P**	GPIO41_PULLEN	**GPIO 41 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO40_P**	GPIO40_PULLEN	**GPIO 40 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO39_P**	GPIO39_PULLEN	**GPIO 39 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO38_P**	GPIO38_PULLEN	**GPIO 38 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO37_P**	GPIO37_PULLEN	**GPIO 37 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO36_P**	GPIO36_PULLEN	**GPIO 36 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO35_P**	GPIO35_PULLEN	**GPIO 35 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO34_P**	GPIO34_PULLEN	**GPIO 34 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO33_P**	GPIO33_PULLEN	**GPIO 33 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO32_P**	GPIO32_PULLEN	**GPIO 32 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005180 **GPIO_PULLEN4** **GPIO Pull-up/Pull-down Enable Register 4** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO63_P	GPIO62_P	GPIO61_P	GPIO60_P	GPIO59_P	GPIO58_P	GPIO57_P	GPIO56_P	GPIO55_P	GPIO54_P	GPIO53_P	GPIO52_P	GPIO51_P	GPIO50_P	GPIO49_P	GPIO48_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO63_P**	GPIO63_PULLEN	**GPIO 63 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO62_P**	GPIO62_PULLEN	**GPIO 62 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO61_P**	GPIO61_PULLEN	**GPIO 61 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO60_P**	GPIO60_PULLEN	**GPIO 60 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**Reserved**	Reserved	**Reserved**
10	**GPIO58_P**	GPIO58_PULLEN	**GPIO 58 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO57_P**	GPIO57_PULLEN	**GPIO 57 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO56_P**	GPIO56_PULLEN	**GPIO 56 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO55_P**	GPIO55_PULLEN	**GPIO 55 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO54_P**	GPIO54_PULLEN	**GPIO 54 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO53_P**	GPIO53_PULLEN	**GPIO 53 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO52_P**	GPIO52_PULLEN	**GPIO 52 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO51_P**	GPIO51_PULLEN	**GPIO 51 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO50_P**	GPIO50_PULLEN	**GPIO 50 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO49_P**	GPIO49_PULLEN	**GPIO 49 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO48_P**	GPIO48_PULLEN	**GPIO 48 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005190 **GPIO_PULL_EN5** **GPIO Pull-up/Pull-down Enable Register 5** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO79_P	GPIO78_P	GPIO77_P	GPIO76_P	GPIO75_P	GPIO74_P	GPIO73_P	GPIO72_P	GPIO71_P	GPIO70_P	GPIO69_P	GPIO68_P	GPIO67_P	GPIO66_P	GPIO65_P	GPIO64_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	GPIO79_P	GPIO79_PULLEN	**GPIO 79 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	GPIO78_P	GPIO78_PULLEN	**GPIO 78 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	GPIO77_P	GPIO77_PULLEN	**GPIO 77 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	GPIO76_P	GPIO76_PULLEN	**GPIO 76 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	GPIO75_P	GPIO75_PULLEN	**GPIO 75 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	GPIO74_P	GPIO74_PULLEN	**GPIO 74 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	GPIO73_P	GPIO73_PULLEN	**GPIO 73 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	GPIO72_P	GPIO72_PULLEN	**GPIO 72 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	GPIO71_P	GPIO71_PULLEN	**GPIO 71 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	GPIO70_P	GPIO70_PULLEN	**GPIO 70 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	GPIO69_P	GPIO69_PULLEN	**GPIO 69 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	GPIO68_P	GPIO68_PULLEN	**GPIO 68 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	GPIO67_P	GPIO67_PULLEN	**GPIO 67 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	GPIO66_P	GPIO66_PULLEN	**GPIO 66 Pull-up/Pull-down Enable** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
1	**GPIO65_P**	GPIO65_PULLEN	**GPIO 65 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO64_P**	GPIO64_PULLEN	**GPIO 64 Pull-up/Pull-down Enable** 0: Disable 1: Enable

100051A0 **GPIO_PULLEN6** **GPIO Pull-up/Pull-down Enable Register 6** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO95_P	GPIO94_P	GPIO93_P	GPIO92_P	GPIO91_P	GPIO90_P	GPIO89_P	GPIO88_P	GPIO87_P	GPIO86_P	GPIO85_P	GPIO84_P	GPIO83_P	GPIO82_P	GPIO81_P	GPIO80_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO95_P**	GPIO95_PULLEN	**GPIO 95 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**Reserved**	Reserved	**Reserved**
13	**Reserved**	Reserved	**Reserved**
12	**Reserved**	Reserved	**Reserved**
11	**Reserved**	Reserved	**Reserved**
10	**GPIO90_P**	GPIO90_PULLEN	**GPIO 90 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO89_P**	GPIO89_PULLEN	**GPIO 89 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO88_P**	GPIO88_PULLEN	**GPIO 88 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO87_P**	GPIO87_PULLEN	**GPIO 87 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO86_P**	GPIO86_PULLEN	**GPIO 86 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO85_P**	GPIO85_PULLEN	**GPIO 85 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO84_P**	GPIO84_PULLEN	**GPIO 84 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO83_**	GPIO83_PULLE	**GPIO 83 Pull-up/Pull-down Enable**

Bit(s)	Mnemonic	Name	Description
	P	N	0: Disable 1: Enable
2	GPIO82_P	GPIO82_PULLEN	GPIO 82 Pull-up/Pull-down Enable 0: Disable 1: Enable
1	GPIO81_P	GPIO81_PULLEN	GPIO 81 Pull-up/Pull-down Enable 0: Disable 1: Enable
0	GPIO80_P	GPIO80_PULLEN	GPIO 80 Pull-up/Pull-down Enable 0: Disable 1: Enable

100051B0 **GPIO_PULLEN7** **GPIO Pull-up/Pull-down Enable Register 7** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO111_P	GPIO110_P	GPIO109_P	GPIO108_P	GPIO107_P	GPIO106_P	GPIO105_P	GPIO104_P	GPIO103_P	GPIO102_P	GPIO101_P	GPIO100_P	GPIO99_P	GPIO98_P	GPIO97_P	GPIO96_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	GPIO111_P	GPIO111_PULLEN	GPIO 111 Pull-up/Pull-down Enable 0: Disable 1: Enable
14	GPIO110_P	GPIO110_PULLEN	GPIO 110 Pull-up/Pull-down Enable 0: Disable 1: Enable
13	GPIO109_P	GPIO109_PULLEN	GPIO 109 Pull-up/Pull-down Enable 0: Disable 1: Enable
12	GPIO108_P	GPIO108_PULLEN	GPIO 108 Pull-up/Pull-down Enable 0: Disable 1: Enable
11	GPIO107_P	GPIO107_PULLEN	GPIO 107 Pull-up/Pull-down Enable 0: Disable 1: Enable
10	GPIO106_P	GPIO106_PULLEN	GPIO 106 Pull-up/Pull-down Enable 0: Disable 1: Enable
9	GPIO105_P	GPIO105_PULLEN	GPIO 105 Pull-up/Pull-down Enable 0: Disable 1: Enable
8	GPIO104_P	GPIO104_PULLEN	GPIO 104 Pull-up/Pull-down Enable 0: Disable 1: Enable
7	GPIO103_P	GPIO103_PULLEN	GPIO 103 Pull-up/Pull-down Enable 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
6	**GPIO102_P**	GPIO102_PULLEN	**GPIO 102 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO101_P**	GPIO101_PULLEN	**GPIO 101 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO100_P**	GPIO100_PULLEN	**GPIO 100 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO99_P**	GPIO99_PULLEN	**GPIO 99 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO98_P**	GPIO98_PULLEN	**GPIO 98 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO97_P**	GPIO97_PULLEN	**GPIO 97 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO96_P**	GPIO96_PULLEN	**GPIO 96 Pull-up/Pull-down Enable** 0: Disable 1: Enable

100051C0 **GPIO_PULLEN8** **GPIO Pull-up/Pull-down Enable Register 8** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO127_P	GPIO126_P	GPIO125_P	GPIO124_P	GPIO123_P	GPIO122_P	GPIO121_P	GPIO120_P	GPIO119_P	GPIO118_P	GPIO117_P	GPIO116_P	GPIO115_P	GPIO114_P	GPIO113_P	GPIO112_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**Reserved**	Reserved	**Reserved**
14	**GPIO126_P**	GPIO126_PULLEN	**GPIO 126 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO125_P**	GPIO125_PULLEN	**GPIO 125 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO124_P**	GPIO124_PULLEN	**GPIO 124 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO123_P**	GPIO123_PULLEN	**GPIO 123 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO122**	GPIO122_PULLE	**GPIO 122 Pull-up/Pull-down Enable**

Bit(s)	Mnemonic	Name	Description
	_P	N	0: Disable 1: Enable
9	GPIO121_P	GPIO121_PULLEN	**GPIO 121 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	GPIO120_P	GPIO120_PULLEN	**GPIO 120 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	GPIO119_P	GPIO119_PULLEN	**GPIO 119 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	GPIO118_P	GPIO118_PULLEN	**GPIO 118 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	GPIO117_P	GPIO117_PULLEN	**GPIO 117 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	GPIO116_P	GPIO116_PULLEN	**GPIO 116 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	GPIO115_P	GPIO115_PULLEN	**GPIO 115 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	GPIO114_P	GPIO114_PULLEN	**GPIO 114 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	GPIO113_P	GPIO113_PULLEN	**GPIO 113 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	GPIO112_P	GPIO112_PULLEN	**GPIO 112 Pull-up/Pull-down Enable** 0: Disable 1: Enable

100051D0 GPIO_PULLEN9 GPIO Pull-up/Pull-down Enable Register 9 FFFF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO143_P	GPIO142_P	GPIO141_P	GPIO140_P	GPIO139_P	GPIO138_P	GPIO137_P	GPIO136_P	GPIO135_P	GPIO134_P	GPIO133_P	GPIO132_P	GPIO131_P	GPIO130_P	GPIO129_P	GPIO128_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0	Reserved	Reserved	Reserved

100051E0 **GPIO_PULL_EN10** **GPIO Pull-up/Pull-down Enable Register 10** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO159_P	GPIO158_P	GPIO157_P	GPIO156_P	GPIO155_P	GPIO154_P	GPIO153_P	GPIO152_P	GPIO151_P	GPIO150_P	GPIO149_P	GPIO148_P	GPIO147_P	GPIO146_P	GPIO145_P	GPIO144_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100051F0 **GPIO_PULL_EN11** **GPIO Pull-up/Pull-down Enable Register 11** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO175_P	GPIO174_P	GPIO173_P	GPIO172_P	GPIO171_P	GPIO170_P	GPIO169_P	GPIO168_P	GPIO167_P	GPIO166_P	GPIO165_P	GPIO164_P	GPIO163_P	GPIO162_P	GPIO161_P	GPIO160_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005200 **GPIO_PULL_EN12** **GPIO Pull-up/Pull-down Enable Register 12** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO191_P	GPIO190_P	GPIO189_P	GPIO188_P	GPIO187_P	GPIO186_P	GPIO185_P	GPIO184_P	GPIO183_P	GPIO182_P	GPIO181_P	GPIO180_P	GPIO179_P	GPIO178_P	GPIO177_P	GPIO176_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO191_P**	GPIO191_PULLEN	**GPIO 191 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO190_P**	GPIO190_PULLEN	**GPIO 190 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO189_P**	GPIO189_PULLEN	**GPIO 189 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO188**	GPIO188_PULLE	**GPIO 188 Pull-up/Pull-down Enable**

Bit(s)	Mnemonic	Name	Description
	_P	N	0: Disable 1: Enable
11:0	**Reserved**	Reserved	**Reserved**

10005210 **GPIO_PULL_EN13** **GPIO Pull-up/Pull-down Enable Register 13** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**GPIO207_P**	**GPIO206_P**	**GPIO205_P**	**GPIO204_P**	**GPIO203_P**	**GPIO202_P**	**GPIO201_P**	**GPIO200_P**	**GPIO199_P**	**GPIO198_P**	**GPIO197_P**	**GPIO196_P**	**GPIO195_P**	**GPIO194_P**	**GPIO193_P**	**GPIO192_P**
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO207_P**	GPIO207_PULLEN	**GPIO 207 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO206_P**	GPIO206_PULLEN	**GPIO 206 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO205_P**	GPIO205_PULLEN	**GPIO 205 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO204_P**	GPIO204_PULLEN	**GPIO 204 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO203_P**	GPIO203_PULLEN	**GPIO 203 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO202_P**	GPIO202_PULLEN	**GPIO 202 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO201_P**	GPIO201_PULLEN	**GPIO 201 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO200_P**	GPIO200_PULLEN	**GPIO 200 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO199_P**	GPIO199_PULLEN	**GPIO 199 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO198_P**	GPIO198_PULLEN	**GPIO 198 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO197_P**	GPIO197_PULLEN	**GPIO 197 Pull-up/Pull-down Enable** 0: Disable 1: Enable

Bit(s)	Mnemonic	Name	Description
4	**GPIO196_P**	GPIO196_PULLEN	**GPIO 196 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO195_P**	GPIO195_PULLEN	**GPIO 195 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO194_P**	GPIO194_PULLEN	**GPIO 194 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO193_P**	GPIO193_PULLEN	**GPIO 193 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO192_P**	GPIO192_PULLEN	**GPIO 192 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005220 **GPIO_PULLEN14** **GPIO Pull-up/Pull-down Enable Register 14** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO223_P	GPIO222_P	GPIO221_P	GPIO220_P	GPIO219_P	GPIO218_P	GPIO217_P	GPIO216_P	GPIO215_P	GPIO214_P	GPIO213_P	GPIO212_P	GPIO211_P	GPIO210_P	GPIO209_P	GPIO208_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:2	**Reserved**	Reserved	**Reserved**
1	**GPIO209_P**	GPIO209_PULLEN	**GPIO 209 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO208_P**	GPIO208_PULLEN	**GPIO 208 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005230 **GPIO_PULLEN15** **GPIO Pull-up/Pull-down Enable Register 15** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO239_P	GPIO238_P	GPIO237_P	GPIO236_P	GPIO235_P	GPIO234_P	GPIO233_P	GPIO232_P	GPIO231_P	GPIO230_P	GPIO229_P	GPIO228_P	GPIO227_P	GPIO226_P	GPIO225_P	GPIO224_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15	**GPIO239_P**	GPIO239_PULLEN	**GPIO 239 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO238_P**	GPIO238_PULLEN	**GPIO 238 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO237_P**	GPIO237_PULLEN	**GPIO 237 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO236_P**	GPIO236_PULLEN	**GPIO 236 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11:0	**Reserved**	Reserved	**Reserved**

10005240 **GPIO_PULLEN16** **GPIO Pull-up/Pull-down Enable Register 16** **FEFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO255_P	GPIO254_P	GPIO253_P	GPIO252_P	GPIO251_P	GPIO250_P	GPIO249_P	GPIO248_P	GPIO247_P	GPIO246_P	GPIO245_P	GPIO244_P	GPIO243_P	GPIO242_P	GPIO241_P	GPIO240_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO255_P**	GPIO255_PULLEN	**GPIO 255 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO254_P**	GPIO254_PULLEN	**GPIO 254 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO253_P**	GPIO253_PULLEN	**GPIO 253 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO252_P**	GPIO252_PULLEN	**GPIO 252 Pull-up/Pull-down Enable** 0: Disable 1: Enable
11	**GPIO251_P**	GPIO251_PULLEN	**GPIO 251 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO250_P**	GPIO250_PULLEN	**GPIO 250 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO249_P**	GPIO249_PULLEN	**GPIO 249 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO248_P**	GPIO248_PULLEN	**GPIO 248 Pull-up/Pull-down Enable** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
7	**GPIO247_P**	GPIO247_PULLEN	**GPIO 247 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO246_P**	GPIO246_PULLEN	**GPIO 246 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO245_P**	GPIO245_PULLEN	**GPIO 245 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO244_P**	GPIO244_PULLEN	**GPIO 244 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO243_P**	GPIO243_PULLEN	**GPIO 243 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO242_P**	GPIO242_PULLEN	**GPIO 242 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO241_P**	GPIO241_PULLEN	**GPIO 241 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO240_P**	GPIO240_PULLEN	**GPIO 240 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005250 **GPIO_PULLEN17** **GPIO Pull-up/Pull-down Enable Register 17** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO271_P	GPIO270_P	GPIO269_P	GPIO268_P	GPIO267_P	GPIO266_P	GPIO265_P	GPIO264_P	GPIO263_P	GPIO262_P	GPIO261_P	GPIO260_P	GPIO259_P	GPIO258_P	GPIO257_P	GPIO256_P
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO271_P**	GPIO271_PULLEN	**GPIO 271 Pull-up/Pull-down Enable** 0: Disable 1: Enable
14	**GPIO270_P**	GPIO270_PULLEN	**GPIO 270 Pull-up/Pull-down Enable** 0: Disable 1: Enable
13	**GPIO269_P**	GPIO269_PULLEN	**GPIO 269 Pull-up/Pull-down Enable** 0: Disable 1: Enable
12	**GPIO268_P**	GPIO268_PULLEN	**GPIO 268 Pull-up/Pull-down Enable** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable
11	**GPIO267_P**	GPIO267_PULLEN	**GPIO 267 Pull-up/Pull-down Enable** 0: Disable 1: Enable
10	**GPIO266_P**	GPIO266_PULLEN	**GPIO 266 Pull-up/Pull-down Enable** 0: Disable 1: Enable
9	**GPIO265_P**	GPIO265_PULLEN	**GPIO 265 Pull-up/Pull-down Enable** 0: Disable 1: Enable
8	**GPIO264_P**	GPIO264_PULLEN	**GPIO 264 Pull-up/Pull-down Enable** 0: Disable 1: Enable
7	**GPIO263_P**	GPIO263_PULLEN	**GPIO 263 Pull-up/Pull-down Enable** 0: Disable 1: Enable
6	**GPIO262_P**	GPIO262_PULLEN	**GPIO 262 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**GPIO261_P**	GPIO261_PULLEN	**GPIO 261 Pull-up/Pull-down Enable** 0: Disable 1: Enable
4	**GPIO260_P**	GPIO260_PULLEN	**GPIO 260 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO259_P**	GPIO259_PULLEN	**GPIO 259 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO258_P**	GPIO258_PULLEN	**GPIO 258 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**GPIO257_P**	GPIO257_PULLEN	**GPIO 257 Pull-up/Pull-down Enable** 0: Disable 1: Enable
0	**GPIO256_P**	GPIO256_PULLEN	**GPIO 256 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005260 **GPIO_PULLEN18** **GPIO Pull-up/Pull-down Enable Register 18** **00FF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									GPIO279_P	GPIO278_P	GPIO277_P	GPIO276_P	GPIO275_P	GPIO274_P	GPIO273_P	GPIO272_P
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
7	**Reserved**	Reserved	**Reserved**
6	**GPIO278_P**	GPIO278_PULLEN	**GPIO 278 Pull-up/Pull-down Enable** 0: Disable 1: Enable
5	**Reserved**	Reserved	**Reserved**
4	**GPIO276_P**	GPIO276_PULLEN	**GPIO 276 Pull-up/Pull-down Enable** 0: Disable 1: Enable
3	**GPIO275_P**	GPIO275_PULLEN	**GPIO 275 Pull-up/Pull-down Enable** 0: Disable 1: Enable
2	**GPIO274_P**	GPIO274_PULLEN	**GPIO 274 Pull-up/Pull-down Enable** 0: Disable 1: Enable
1	**Reserved**	Reserved	**Reserved**
0	**GPIO272_P**	GPIO272_PULLEN	**GPIO 272 Pull-up/Pull-down Enable** 0: Disable 1: Enable

10005280 **GPIO_PULLSEL1** **GPIO Pull-up/Pull-down Selection Register 1** **1050**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO15_U	GPIO14_U	GPIO13_U	GPIO12_U	GPIO11_U	GPIO10_U	GPIO9_U	GPIO8_U	GPIO7_U	GPIO6_U	GPIO5_U	GPIO4_U	GPIO3_U	GPIO2_U	GPIO1_U	GPIO0_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO15_U**	GPIO15_PULLSEL	**GPIO 15 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO14_U**	GPIO14_PULLSEL	**GPIO 14 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO13_U**	GPIO13_PULLSEL	**GPIO 13 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO12_U**	GPIO12_PULLSEL	**GPIO 12 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO11_U**	GPIO11_PULLSEL	**GPIO 11 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO10_U**	GPIO10_PULLSEL	**GPIO 10 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

Bit(s)	Mnemonic	Name	Description
9	**GPIO9_U**	GPIO9_PULLSEL	**GPIO 9 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO8_U**	GPIO8_PULLSEL	**GPIO 8 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO7_U**	GPIO7_PULLSEL	**GPIO 7 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO6_U**	GPIO6_PULLSEL	**GPIO 6 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO5_U**	GPIO5_PULLSEL	**GPIO 5 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO4_U**	GPIO4_PULLSEL	**GPIO 4 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO3_U**	GPIO3_PULLSEL	**GPIO 3 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO2_U**	GPIO2_PULLSEL	**GPIO 2 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO1_U**	GPIO1_PULLSEL	**GPIO 1 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO0_U**	GPIO0_PULLSEL	**GPIO 0 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

10005290 **GPIO_PULLSEL2** **GPIO Pull-up/Pull-down Selection Register 2** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO31_U	GPIO30_U	GPIO29_U	GPIO28_U	GPIO27_U	GPIO26_U	GPIO25_U	GPIO24_U	GPIO23_U	GPIO22_U	GPIO21_U	GPIO20_U	GPIO19_U	GPIO18_U	GPIO17_U	GPIO16_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO31_U**	GPIO31_PULLSEL	**GPIO 31 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO30_U**	GPIO30_PULLSEL	**GPIO 30 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

Bit(s)	Mnemonic	Name	Description
13	**GPIO29_U**	GPIO29_PULLSEL	**GPIO 29 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO28_U**	GPIO28_PULLSEL	**GPIO 28 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO27_U**	GPIO27_PULLSEL	**GPIO 27 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO26_U**	GPIO26_PULLSEL	**GPIO 26 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO25_U**	GPIO25_PULLSEL	**GPIO 25 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO24_U**	GPIO24_PULLSEL	**GPIO 24 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO23_U**	GPIO23_PULLSEL	**GPIO 23 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO22_U**	GPIO22_PULLSEL	**GPIO 22 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO21_U**	GPIO21_PULLSEL	**GPIO 21 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO20_U**	GPIO20_PULLSEL	**GPIO 20 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO19_U**	GPIO19_PULLSEL	**GPIO 19 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO18_U**	GPIO18_PULLSEL	**GPIO 18 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO17_U**	GPIO17_PULLSEL	**GPIO 17 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO16_U**	GPIO16_PULLSEL	**GPIO 16 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052A0 **GPIO_PULLSEL3** **GPIO Pull-up/Pull-down Selection Register 3** **0780**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**GPIO47_**	**GPIO46_**	**GPIO45_**	**GPIO44_**	**GPIO43_**	**GPIO42_**	**GPIO41_U**	**GPIO40_**	**GPIO39_**	**GPIO38_**	**GPIO37_**	**GPIO36_**	**GPIO35_**	**GPIO34_**	**GPIO33_**	**GPIO32_**

100052A0 **GPIO_PULL_SEL3** **GPIO Pull-up/Pull-down Selection Register 3** **0780**

	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U	U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	GPIO47_U	GPIO47_PULLSEL	**GPIO 47 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	GPIO46_U	GPIO46_PULLSEL	**GPIO 46 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	GPIO45_U	GPIO45_PULLSEL	**GPIO 45 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	GPIO44_U	GPIO44_PULLSEL	**GPIO 44 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	GPIO43_U	GPIO43_PULLSEL	**GPIO 43 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	GPIO42_U	GPIO42_PULLSEL	**GPIO 42 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	GPIO41_U	GPIO41_PULLSEL	**GPIO 41 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	GPIO40_U	GPIO40_PULLSEL	**GPIO 40 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	GPIO39_U	GPIO39_PULLSEL	**GPIO 39 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	GPIO38_U	GPIO38_PULLSEL	**GPIO 38 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	GPIO37_U	GPIO37_PULLSEL	**GPIO 37 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	GPIO36_U	GPIO36_PULLSEL	**GPIO 36 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	GPIO35_U	GPIO35_PULLSEL	**GPIO 35 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	GPIO34_U	GPIO34_PULLSEL	**GPIO 34 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	GPIO33_U	GPIO33_PULLSEL	**GPIO 33 Pull-up/Pull-down Selection** 0: Pull-down

Bit(s)	Mnemonic	Name	Description
			1: Pull-up
0	**GPIO32_U**	GPIO32_PULLSEL	**GPIO 32 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052B0 **GPIO_PULLSEL4** **GPIO Pull-up/Pull-down Selection Register 4** **0600**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO63_U	GPIO62_U	GPIO61_U	GPIO60_U	GPIO59_U	GPIO58_U	GPIO57_U	GPIO56_U	GPIO55_U	GPIO54_U	GPIO53_U	GPIO52_U	GPIO51_U	GPIO50_U	GPIO49_U	GPIO48_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO63_U**	GPIO63_PULLSEL	**GPIO 63 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO62_U**	GPIO62_PULLSEL	**GPIO 62 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO61_U**	GPIO61_PULLSEL	**GPIO 61 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO60_U**	GPIO60_PULLSEL	**GPIO 60 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**Reserved**	Reserved	**Reserved**
10	**GPIO58_U**	GPIO58_PULLSEL	**GPIO 58 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO57_U**	GPIO57_PULLSEL	**GPIO 57 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO56_U**	GPIO56_PULLSEL	**GPIO 56 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO55_U**	GPIO55_PULLSEL	**GPIO 55 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO54_U**	GPIO54_PULLSEL	**GPIO 54 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO53_U**	GPIO53_PULLSEL	**GPIO 53 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO52_**	GPIO52_PULLSE	**GPIO 52 Pull-up/Pull-down Selection**

Bit(s)	Mnemonic	Name	Description
	U	L	0: Pull-down 1: Pull-up
3	**GPIO51_U**	GPIO51_PULLSEL	**GPIO 51 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO50_U**	GPIO50_PULLSEL	**GPIO 50 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO49_U**	GPIO49_PULLSEL	**GPIO 49 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO48_U**	GPIO48_PULLSEL	**GPIO 48 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052C0 **GPIO_PULLSEL5** **GPIO Pull-up/Pull-down Selection Register 5** **F800**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO79_U	GPIO78_U	GPIO77_U	GPIO76_U	GPIO75_U	GPIO74_U	GPIO73_U	GPIO72_U	GPIO71_U	GPIO70_U	GPIO69_U	GPIO68_U	GPIO67_U	GPIO66_U	GPIO65_U	GPIO64_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO79_U**	GPIO79_PULLSEL	**GPIO 79 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO78_U**	GPIO78_PULLSEL	**GPIO 78 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO77_U**	GPIO77_PULLSEL	**GPIO 77 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO76_U**	GPIO76_PULLSEL	**GPIO 76 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO75_U**	GPIO75_PULLSEL	**GPIO 75 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO74_U**	GPIO74_PULLSEL	**GPIO 74 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO73_U**	GPIO73_PULLSEL	**GPIO 73 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO72_U**	GPIO72_PULLSEL	**GPIO 72 Pull-up/Pull-down Selection** 0: Pull-down

Bit(s)	Mnemonic	Name	Description
			1: Pull-up
7	**GPIO71_U**	GPIO71_PULLSEL	**GPIO 71 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO70_U**	GPIO70_PULLSEL	**GPIO 70 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO69_U**	GPIO69_PULLSEL	**GPIO 69 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO68_U**	GPIO68_PULLSEL	**GPIO 68 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO67_U**	GPIO67_PULLSEL	**GPIO 67 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO66_U**	GPIO66_PULLSEL	**GPIO 66 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO65_U**	GPIO65_PULLSEL	**GPIO 65 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO64_U**	GPIO64_PULLSEL	**GPIO 64 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052D0 **GPIO_PULLSEL6** **GPIO Pull-up/Pull-down Selection Register 6** **0001**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO95_U	GPIO94_U	GPIO93_U	GPIO92_U	GPIO91_U	GPIO90_U	GPIO89_U	GPIO88_U	GPIO87_U	GPIO86_U	GPIO85_U	GPIO84_U	GPIO83_U	GPIO82_U	GPIO81_U	GPIO80_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15	**GPIO95_U**	GPIO95_PULLSEL	**GPIO 95 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**Reserved**	Reserved	**Reserved**
13	**Reserved**	Reserved	**Reserved**
12	**Reserved**	Reserved	**Reserved**
11	**Reserved**	Reserved	**Reserved**
10	**GPIO90_U**	GPIO90_PULLSEL	**GPIO 90 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO89_**	GPIO89_PULLS	**GPIO 89 Pull-up/Pull-down Selection**

Bit(s)	Mnemonic	Name	Description
	U	EL	0: Pull-down 1: Pull-up
8	**GPIO88_U**	GPIO88_PULLSEL	**GPIO 88 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO87_U**	GPIO87_PULLSEL	**GPIO 87 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO86_U**	GPIO86_PULLSEL	**GPIO 86 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO85_U**	GPIO85_PULLSEL	**GPIO 85 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO84_U**	GPIO84_PULLSEL	**GPIO 84 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO83_U**	GPIO83_PULLSEL	**GPIO 83 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO82_U**	GPIO82_PULLSEL	**GPIO 82 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO81_U**	GPIO81_PULLSEL	**GPIO 81 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO80_U**	GPIO80_PULLSEL	**GPIO 80 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052E0 **GPIO_PULLSEL7** **GPIO Pull-up/Pull-down Selection Register 7** **FA00**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO111_U	GPIO110_U	GPIO109_U	GPIO108_U	GPIO107_U	GPIO106_U	GPIO105_U	GPIO104_U	GPIO103_U	GPIO102_U	GPIO101_U	GPIO100_U	GPIO99_U	GPIO98_U	GPIO97_U	GPIO96_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	0	1	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO111_U**	GPIO111_PULLSEL	**GPIO 111 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO110_U**	GPIO110_PULLSEL	**GPIO 110 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO109_U**	GPIO109_PULLSEL	**GPIO 109 Pull-up/Pull-down Selection** 0: Pull-down

Bit(s)	Mnemonic	Name	Description
			1: Pull-up
12	**GPIO108_U**	GPIO108_PULLSEL	**GPIO 108 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO107_U**	GPIO107_PULLSEL	**GPIO 107 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO106_U**	GPIO106_PULLSEL	**GPIO 106 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO105_U**	GPIO105_PULLSEL	**GPIO 105 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO104_U**	GPIO104_PULLSEL	**GPIO 104 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO103_U**	GPIO103_PULLSEL	**GPIO 103 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO102_U**	GPIO102_PULLSEL	**GPIO 102 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO101_U**	GPIO101_PULLSEL	**GPIO 101 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO100_U**	GPIO100_PULLSEL	**GPIO 100 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO99_U**	GPIO99_PULLSEL	**GPIO 99 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO98_U**	GPIO98_PULLSEL	**GPIO 98 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO97_U**	GPIO97_PULLSEL	**GPIO 97 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO96_U**	GPIO96_PULLSEL	**GPIO 96 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100052F0 **GPIO_PULLSEL8** **GPIO Pull-up/Pull-down Selection Register 8** **3FDF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO127_U	GPIO126_U	GPIO125_U	GPIO124_U	GPIO123_U	GPIO122_U	GPIO121_U	GPIO120_U	GPIO119_U	GPIO118_U	GPIO117_U	GPIO116_U	GPIO115_U	GPIO114_U	GPIO113_U	GPIO112_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15	**Reserved**	Reserved	**Reserved**
14	**GPIO126_U**	GPIO126_PULLSEL	**GPIO 126 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO125_U**	GPIO125_PULLSEL	**GPIO 125 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO124_U**	GPIO124_PULLSEL	**GPIO 124 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO123_U**	GPIO123_PULLSEL	**GPIO 123 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO122_U**	GPIO122_PULLSEL	**GPIO 122 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO121_U**	GPIO121_PULLSEL	**GPIO 121 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO120_U**	GPIO120_PULLSEL	**GPIO 120 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO119_U**	GPIO119_PULLSEL	**GPIO 119 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO118_U**	GPIO118_PULLSEL	**GPIO 118 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO117_U**	GPIO117_PULLSEL	**GPIO 117 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO116_U**	GPIO116_PULLSEL	**GPIO 116 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO115_U**	GPIO115_PULLSEL	**GPIO 115 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO114_U**	GPIO114_PULLSEL	**GPIO 114 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO113_U**	GPIO113_PULLSEL	**GPIO 113 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO112_U**	GPIO112_PULLSEL	**GPIO 112 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

10005300 **GPIO_PULL_SEL9** **GPIO Pull-up/Pull-down Selection Register 9** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO143_U	GPIO142_U	GPIO141_U	GPIO140_U	GPIO139_U	GPIO138_U	GPIO137_U	GPIO136_U	GPIO135_U	GPIO134_U	GPIO133_U	GPIO132_U	GPIO131_U	GPIO130_U	GPIO129_U	GPIO128_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005310 **GPIO_PULL_SEL10** **GPIO Pull-up/Pull-down Selection Register 10** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO159_U	GPIO158_U	GPIO157_U	GPIO156_U	GPIO155_U	GPIO154_U	GPIO153_U	GPIO152_U	GPIO151_U	GPIO150_U	GPIO149_U	GPIO148_U	GPIO147_U	GPIO146_U	GPIO145_U	GPIO144_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005320 **GPIO_PULL_SEL11** **GPIO Pull-up/Pull-down Selection Register 11** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO175_U	GPIO174_U	GPIO173_U	GPIO172_U	GPIO171_U	GPIO170_U	GPIO169_U	GPIO168_U	GPIO167_U	GPIO166_U	GPIO165_U	GPIO164_U	GPIO163_U	GPIO162_U	GPIO161_U	GPIO160_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005330 **GPIO_PULL_SEL12** **GPIO Pull-up/Pull-down Selection Register 12** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO191_	GPIO190_	GPIO189	GPIO188	GPIO187_	GPIO186	GPIO185	GPIO184	GPIO183	GPIO182	GPIO181_	GPIO180	GPIO179_	GPIO178_	GPIO177_	GPIO176_

10005330 **GPIO_PULL SEL12** **GPIO Pull-up/Pull-down Selection Register 12** **0000**

	U	U	_U	_U	U	_U	_U	_U	_U	_U	U	_U	U	U	U	U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO191_U**	GPIO191_PULLSEL	**GPIO 191 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO190_U**	GPIO190_PULLSEL	**GPIO 190 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO189_U**	GPIO189_PULLSEL	**GPIO 189 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO188_U**	GPIO188_PULLSEL	**GPIO 188 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11:0	**Reserved**	Reserved	**Reserved**

10005340 **GPIO_PULL SEL13** **GPIO Pull-up/Pull-down Selection Register 13** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**GPIO207_U**	**GPIO206_U**	**GPIO205_U**	**GPIO204_U**	**GPIO203_U**	**GPIO202_U**	**GPIO201_U**	**GPIO200_U**	**GPIO199_U**	**GPIO198_U**	**GPIO197_U**	**GPIO196_U**	**GPIO195_U**	**GPIO194_U**	**GPIO193_U**	**GPIO192_U**
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO207_U**	GPIO207_PULLSEL	**GPIO 207 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO206_U**	GPIO206_PULLSEL	**GPIO 206 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO205_U**	GPIO205_PULLSEL	**GPIO 205 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO204_U**	GPIO204_PULLSEL	**GPIO 204 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO203_U**	GPIO203_PULLSEL	**GPIO 203 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO202_U**	GPIO202_PULLSEL	**GPIO 202 Pull-up/Pull-down Selection** 0: Pull-down

Bit(s)	Mnemonic	Name	Description
			1: Pull-up
9	**GPIO201_U**	GPIO201_PULLSEL	**GPIO 201 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO200_U**	GPIO200_PULLSEL	**GPIO 200 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO199_U**	GPIO199_PULLSEL	**GPIO 199 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO198_U**	GPIO198_PULLSEL	**GPIO 198 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO197_U**	GPIO197_PULLSEL	**GPIO 197 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO196_U**	GPIO196_PULLSEL	**GPIO 196 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO195_U**	GPIO195_PULLSEL	**GPIO 195 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO194_U**	GPIO194_PULLSEL	**GPIO 194 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO193_U**	GPIO193_PULLSEL	**GPIO 193 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO192_U**	GPIO192_PULLSEL	**GPIO 192 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

10005350 **GPIO_PULLSEL14** **GPIO Pull-up/Pull-down Selection Register 14** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO223_U	GPIO222_U	GPIO221_U	GPIO220_U	GPIO219_U	GPIO218_U	GPIO217_U	GPIO216_U	GPIO215_U	GPIO214_U	GPIO213_U	GPIO212_U	GPIO211_U	GPIO210_U	GPIO209_U	GPIO208_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:2	**Reserved**	Reserved	**Reserved**
1	**GPIO209_U**	GPIO209_PULLSEL	**GPIO 209 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO208**	GPIO208_PULLS	**GPIO 208 Pull-up/Pull-down Selection**

Bit(s)	Mnemonic	Name	Description
	_U	EL	0: Pull-down 1: Pull-up

10005360 **GPIO_PULL SEL15** **GPIO Pull-up/Pull-down Selection Register 15** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO239_U	GPIO238_U	GPIO237_U	GPIO236_U	GPIO235_U	GPIO234_U	GPIO233_U	GPIO232_U	GPIO231_U	GPIO230_U	GPIO229_U	GPIO228_U	GPIO227_U	GPIO226_U	GPIO225_U	GPIO224_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO239_U**	GPIO239_PULLSEL	**GPIO 239 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO238_U**	GPIO238_PULLSEL	**GPIO 238 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO237_U**	GPIO237_PULLSEL	**GPIO 237 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	**GPIO236_U**	GPIO236_PULLSEL	**GPIO 236 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11:0	**Reserved**	Reserved	**Reserved**

10005370 **GPIO_PULL SEL16** **GPIO Pull-up/Pull-down Selection Register 16** **FEB0**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO255_U	GPIO254_U	GPIO253_U	GPIO252_U	GPIO251_U	GPIO250_U	GPIO249_U	GPIO248_U	GPIO247_U	GPIO246_U	GPIO245_U	GPIO244_U	GPIO243_U	GPIO242_U	GPIO241_U	GPIO240_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	0	1	0	1	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPIO255_U**	GPIO255_PULLSEL	**GPIO 255 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	**GPIO254_U**	GPIO254_PULLSEL	**GPIO 254 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	**GPIO253**	GPIO253_PULLS	**GPIO 253 Pull-up/Pull-down Selection**

Bit(s)	Mnemonic	Name	Description
	_U	EL	0: Pull-down 1: Pull-up
12	**GPIO252_U**	GPIO252_PULLSEL	**GPIO 252 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	**GPIO251_U**	GPIO251_PULLSEL	**GPIO 251 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	**GPIO250_U**	GPIO250_PULLSEL	**GPIO 250 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	**GPIO249_U**	GPIO249_PULLSEL	**GPIO 249 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	**GPIO248_U**	GPIO248_PULLSEL	**GPIO 248 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	**GPIO247_U**	GPIO247_PULLSEL	**GPIO 247 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	**GPIO246_U**	GPIO246_PULLSEL	**GPIO 246 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**GPIO245_U**	GPIO245_PULLSEL	**GPIO 245 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	**GPIO244_U**	GPIO244_PULLSEL	**GPIO 244 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO243_U**	GPIO243_PULLSEL	**GPIO 243 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO242_U**	GPIO242_PULLSEL	**GPIO 242 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**GPIO241_U**	GPIO241_PULLSEL	**GPIO 241 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	**GPIO240_U**	GPIO240_PULLSEL	**GPIO 240 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

10005380 **GPIO_PULLSEL17** **GPIO Pull-up/Pull-down Selection Register 17** **0017**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO271_U	GPIO270_U	GPIO269_U	GPIO268_U	GPIO267_U	GPIO266_U	GPIO265_U	GPIO264_U	GPIO263_U	GPIO262_U	GPIO261_U	GPIO260_U	GPIO259_U	GPIO258_U	GPIO257_U	GPIO256_U
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

10005380	GPIO_PULL SEL17	GPIO Pull-up/Pull-down Selection Register 17	0017

Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1

Bit(s)	Mnemonic	Name	Description
15	GPIO271_U	GPIO271_PULLSEL	**GPIO 271 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
14	GPIO270_U	GPIO270_PULLSEL	**GPIO 270 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
13	GPIO269_U	GPIO269_PULLSEL	**GPIO 269 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
12	GPIO268_U	GPIO268_PULLSEL	**GPIO 268 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
11	GPIO267_U	GPIO267_PULLSEL	**GPIO 267 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
10	GPIO266_U	GPIO266_PULLSEL	**GPIO 266 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
9	GPIO265_U	GPIO265_PULLSEL	**GPIO 265 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
8	GPIO264_U	GPIO264_PULLSEL	**GPIO 264 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
7	GPIO263_U	GPIO263_PULLSEL	**GPIO 263 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
6	GPIO262_U	GPIO262_PULLSEL	**GPIO 262 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	GPIO261_U	GPIO261_PULLSEL	**GPIO 261 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
4	GPIO260_U	GPIO260_PULLSEL	**GPIO 260 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	GPIO259_U	GPIO259_PULLSEL	**GPIO 259 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	GPIO258_U	GPIO258_PULLSEL	**GPIO 258 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	GPIO257_U	GPIO257_PULLSEL	**GPIO 257 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
0	GPIO256	GPIO256_PULLS	**GPIO 256 Pull-up/Pull-down Selection**

Bit(s)	Mnemonic	Name	Description
	_U	EL	0: Pull-down 1: Pull-up

10005390 **GPIO_PULLSEL18** **GPIO Pull-up/Pull-down Selection Register 18** **0040**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									GPIO279_U	GPIO278_U	GPIO277_U	GPIO276_U	GPIO275_U	GPIO274_U	GPIO273_U	GPIO272_U
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	1	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	**Reserved**	Reserved	**Reserved**
6	**GPIO278_U**	GPIO278_PULLSEL	**GPIO 278 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
5	**Reserved**	Reserved	**Reserved**
4	**GPIO276_U**	GPIO276_PULLSEL	**GPIO 276 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
3	**GPIO275_U**	GPIO275_PULLSEL	**GPIO 275 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
2	**GPIO274_U**	GPIO274_PULLSEL	**GPIO 274 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up
1	**Reserved**	Reserved	**Reserved**
0	**GPIO272_U**	GPIO272_PULLSEL	**GPIO 272 Pull-up/Pull-down Selection** 0: Pull-down 1: Pull-up

100053A0 **MSDC3_CTRL7** **MSDC 3 RCLK Pad Control Register 3** **0414**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3RCK_BACKUP1				MS3RCK_SMT	MS3RCK_R0	MS3RCK_R1	MS3RCK_PUPD	MS3RCK_BACKUP0			MS3RCK_IES	MS3RCK_SR	MS3RCK_E8	MS3RCK_E4	MS3RCK_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15:12		MS3RCK_BACKUP1	**Reserved**
11		MS3RCK_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS3RCK_R0	**10K resistot control**
9		MS3RCK_R1	**50K resistor control**
8		MS3RCK_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MS3RCK_BACKUP0	**Reserved**
4		MS3RCK_IES	**Input enable control**
3		MS3RCK_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS3RCK_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005410 BIAS_CTRL3 18OD33 IO Group BIAS Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GATE_MC02_CTRL				GATE_MC01_CTRL				GATE_NORB_CTRL				GATE_NOR_CTRL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:12		GATE_MC02_CTRL	**BIAS PAD TUNE Control**
11:8		GATE_MC01_CTRL	**BIAS PAD TUNE Control**
7:4		GATE_NORB_CTRL	**BIAS PAD TUNE Control**
3:0		GATE_NOR_CTRL	**BIAS PAD TUNE Control**

10005420 BIAS_CTRL4 18OD33 IO Group BIAS Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BIAS_CTRL4_BACKUP0								GATE_MC1_CTRL				GATE_MC03_CTRL			

10005420 BIAS_CTRL4 18OD33 IO Group BIAS Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Type	RW								RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8		BIAS_CTRL4_BACKUP0	Reserved
7:4		GATE_MC1_CTRL	BIAS PAD TUNE Control
3:0		GATE_MC03_CTRL	BIAS PAD TUNE Control

10005430 MSDC3_CTRL8 MSDC3 RCLK SEL Control Register 00AC

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3RCLK_BACKUP						MS3RCLK_RDSEL						MS3RCLK_TDSEL			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0

Bit(s)	Mnemonic	Name	Description
15:10		MS3RCLK_BACKUP	Reserved
9:4		MS3RCLK_RDSEL	BIAS PAD TUNE Control
3:0		MS3RCLK_TDSEL	BIAS PAD TUNE Control

10005440 OD33_CTRL11 18OD33 IO Group TDSEL/RDSEL Control Register 11 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP11						OD33_RDSEL11						OD33_TDSEL11			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP11	OD33 Control Reserve Registers
9:4		OD33_RDSEL11	RDSEL control register for eint17_tdsel
3:0		OD33_TDSEL11	TDSEL control register for eint17_tdsel

10005450 **OD33_CTRL12** **18OD33 IO Group TDSEL/RDSEL Control Register 12** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP12						OD33_RDSEL12						OD33_TDSEL12			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP12	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL12	**RDSEL control register for eint17_tdsel**
3:0		OD33_TDSEL12	**TDSEL control register for eint17_tdsel**

10005460 **OD33_CTRL13** **18OD33 IO Group TDSEL/RDSEL Control Register 13** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP13						OD33_RDSEL13						OD33_TDSEL13			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP13	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL13	**RDSEL control register for uart2_tdsel, pcm_tdsel, einto_tdsel and eint5_tdsel**
3:0		OD33_TDSEL13	**TDSEL control register for uart2_tdsel, pcm_tdsel, einto_tdsel and eint5_tdsel**

10005470 **OD33_CTRL14** **18OD33 IO Group TDSEL/RDSEL Control Register 14** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP14						OD33_RDSEL14						OD33_TDSEL14			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP14	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL14	**RDSEL control register for eint12_tdsel**
3:0		OD33_TDSEL14	**TDSEL control register for eint12_tdsel**

100054C0 OD33_CTRL8 18OD33 IO Group TDSEL/RDSEL Control Register 8 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP8						OD33_RDSEL8						OD33_TDSEL8			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP8	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL8	**RDSEL control register for eint17_tdsel**
3:0		OD33_TDSEL8	**TDSEL control register for eint17_tdsel**

100054D0 OD33_CTRL9 18OD33 IO Group TDSEL/RDSEL Control Register 9 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP9						OD33_RDSEL9						OD33_TDSEL9			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP9	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL9	**RDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**
3:0		OD33_TDSEL9	**TDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**

100054E0 OD33_CTRL10 18OD33 IO Group TDSEL/RDSEL Control Register 10 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP10						OD33_RDSEL10						OD33_TDSEL10			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP10	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL10	**RDSEL control register for eint12_tdsel**
3:0		OD33_TDSEL10	**TDSEL control register for eint12_tdsel**

10005500 **GPIO_DOUT1** **GPIO Data Output Register 1** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO15	GPO14	GPO13	GPO12	GPO11	GPO10	GPO9	GPO8	GPO7	GPO6	GPO5	GPO4	GPO3	GPO2	GPO1	GPO0
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO15**	GPIO15_DOUT	**GPIO 15 Data Output Value** 0: Output 0 1: Output 1
14	**GPO14**	GPIO14_DOUT	**GPIO 14 Data Output Value** 0: Output 0 1: Output 1
13	**GPO13**	GPIO13_DOUT	**GPIO 13 Data Output Value** 0: Output 0 1: Output 1
12	**GPO12**	GPIO12_DOUT	**GPIO 12 Data Output Value** 0: Output 0 1: Output 1
11	**GPO11**	GPIO11_DOUT	**GPIO 11 Data Output Value** 0: Output 0 1: Output 1
10	**GPO10**	GPIO10_DOUT	**GPIO 10 Data Output Value** 0: Output 0 1: Output 1
9	**GPO9**	GPIO9_DOUT	**GPIO 9 Data Output Value** 0: Output 0 1: Output 1
8	**GPO8**	GPIO8_DOUT	**GPIO 8 Data Output Value** 0: Output 0 1: Output 1
7	**GPO7**	GPIO7_DOUT	**GPIO 7 Data Output Value** 0: Output 0 1: Output 1
6	**GPO6**	GPIO6_DOUT	**GPIO 6 Data Output Value** 0: Output 0 1: Output 1
5	**GPO5**	GPIO5_DOUT	**GPIO 5 Data Output Value** 0: Output 0 1: Output 1
4	**GPO4**	GPIO4_DOUT	**GPIO 4 Data Output Value** 0: Output 0 1: Output 1
3	**GPO3**	GPIO3_DOUT	**GPIO 3 Data Output Value** 0: Output 0 1: Output 1
2	**GPO2**	GPIO2_DOUT	**GPIO 2 Data Output Value** 0: Output 0 1: Output 1

Bit(s)	Mnemonic	Name	Description
1	**GPO1**	GPIO1_DOUT	**GPIO 1 Data Output Value** 0: Output 0 1: Output 1
0	**GPO0**	GPIO0_DOUT	**GPIO 0 Data Output Value** 0: Output 0 1: Output 1

10005510 GPIO_DOUT2 GPIO Data Output Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO31	GPO30	GPO29	GPO28	GPO27	GPO26	GPO25	GPO24	GPO23	GPO22	GPO21	GPO20	GPO19	GPO18	GPO17	GPO16
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO31**	GPIO31_DOUT	**GPIO 31 Data Output Value** 0: Output 0 1: Output 1
14	**GPO30**	GPIO30_DOUT	**GPIO 30 Data Output Value** 0: Output 0 1: Output 1
13	**GPO29**	GPIO29_DOUT	**GPIO 29 Data Output Value** 0: Output 0 1: Output 1
12	**GPO28**	GPIO28_DOUT	**GPIO 28 Data Output Value** 0: Output 0 1: Output 1
11	**GPO27**	GPIO27_DOUT	**GPIO 27 Data Output Value** 0: Output 0 1: Output 1
10	**GPO26**	GPIO26_DOUT	**GPIO 26 Data Output Value** 0: Output 0 1: Output 1
9	**GPO25**	GPIO25_DOUT	**GPIO 25 Data Output Value** 0: Output 0 1: Output 1
8	**GPO24**	GPIO24_DOUT	**GPIO 24 Data Output Value** 0: Output 0 1: Output 1
7	**GPO23**	GPIO23_DOUT	**GPIO 23 Data Output Value** 0: Output 0 1: Output 1
6	**GPO22**	GPIO22_DOUT	**GPIO 22 Data Output Value** 0: Output 0 1: Output 1
5	**GPO21**	GPIO21_DOUT	**GPIO 21 Data Output Value**

Bit(s)	Mnemonic	Name	Description
			0: Output 0 1: Output 1
4	**GPO20**	GPIO20_DOUT	**GPIO 20 Data Output Value** 0: Output 0 1: Output 1
3	**GPO19**	GPIO19_DOUT	**GPIO 19 Data Output Value** 0: Output 0 1: Output 1
2	**GPO18**	GPIO18_DOUT	**GPIO 18 Data Output Value** 0: Output 0 1: Output 1
1	**GPO17**	GPIO17_DOUT	**GPIO 17 Data Output Value** 0: Output 0 1: Output 1
0	**GPO16**	GPIO16_DOUT	**GPIO 16 Data Output Value** 0: Output 0 1: Output 1

10005520 **GPIO_DOUT3** **GPIO Data Output Register 3** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO47	GPO46	GPO45	GPO44	GPO43	GPO42	GPO41	GPO40	GPO39	GPO38	GPO37	GPO36	GPO35	GPO34	GPO33	GPO32
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO47**	GPIO47_DOUT	**GPIO 47 Data Output Value** 0: Output 0 1: Output 1
14	**GPO46**	GPIO46_DOUT	**GPIO 46 Data Output Value** 0: Output 0 1: Output 1
13	**GPO45**	GPIO45_DOUT	**GPIO 45 Data Output Value** 0: Output 0 1: Output 1
12	**GPO44**	GPIO44_DOUT	**GPIO 44 Data Output Value** 0: Output 0 1: Output 1
11	**GPO43**	GPIO43_DOUT	**GPIO 43 Data Output Value** 0: Output 0 1: Output 1
10	**GPO42**	GPIO42_DOUT	**GPIO 42 Data Output Value** 0: Output 0 1: Output 1
9	**GPO41**	GPIO41_DOUT	**GPIO 41 Data Output Value** 0: Output 0

Bit(s)	Mnemonic	Name	Description
			1: Output 1
8	**GPO40**	GPIO40_DOUT	**GPIO 40 Data Output Value** 0: Output 0 1: Output 1
7	**GPO39**	GPIO39_DOUT	**GPIO 39 Data Output Value** 0: Output 0 1: Output 1
6	**GPO38**	GPIO38_DOUT	**GPIO 38 Data Output Value** 0: Output 0 1: Output 1
5	**GPO37**	GPIO37_DOUT	**GPIO 37 Data Output Value** 0: Output 0 1: Output 1
4	**GPO36**	GPIO36_DOUT	**GPIO 36 Data Output Value** 0: Output 0 1: Output 1
3	**GPO35**	GPIO35_DOUT	**GPIO 35 Data Output Value** 0: Output 0 1: Output 1
2	**GPO34**	GPIO34_DOUT	**GPIO 34 Data Output Value** 0: Output 0 1: Output 1
1	**GPO33**	GPIO33_DOUT	**GPIO 33 Data Output Value** 0: Output 0 1: Output 1
0	**GPO32**	GPIO32_DOUT	**GPIO 32 Data Output Value** 0: Output 0 1: Output 1

10005530 **GPIO_DOUT4** **GPIO Data Output Register 4** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO63	GPO62	GPO61	GPO60	GPO59	GPO58	GPO57	GPO56	GPO55	GPO54	GPO53	GPO52	GPO51	GPO50	GPO49	GPO48
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO63**	GPIO63_DOUT	**GPIO 63 Data Output Value** 0: Output 0 1: Output 1
14	**GPO62**	GPIO62_DOUT	**GPIO 62 Data Output Value** 0: Output 0 1: Output 1
13	**GPO61**	GPIO61_DOUT	**GPIO 61 Data Output Value** 0: Output 0 1: Output 1

Bit(s)	Mnemonic	Name	Description
12	**GPO60**	GPIO60_DOUT	**GPIO 60 Data Output Value** 0: Output 0 1: Output 1
11	**Reserved**	Reserved	**Reserved**
10	**GPO58**	GPIO58_DOUT	**GPIO 58 Data Output Value** 0: Output 0 1: Output 1
9	**GPO57**	GPIO57_DOUT	**GPIO 57 Data Output Value** 0: Output 0 1: Output 1
8	**GPO56**	GPIO56_DOUT	**GPIO 56 Data Output Value** 0: Output 0 1: Output 1
7	**GPO55**	GPIO55_DOUT	**GPIO 55 Data Output Value** 0: Output 0 1: Output 1
6	**GPO54**	GPIO54_DOUT	**GPIO 54 Data Output Value** 0: Output 0 1: Output 1
5	**GPO53**	GPIO53_DOUT	**GPIO 53 Data Output Value** 0: Output 0 1: Output 1
4	**GPO52**	GPIO52_DOUT	**GPIO 52 Data Output Value** 0: Output 0 1: Output 1
3	**GPO51**	GPIO51_DOUT	**GPIO 51 Data Output Value** 0: Output 0 1: Output 1
2	**GPO50**	GPIO50_DOUT	**GPIO 50 Data Output Value** 0: Output 0 1: Output 1
1	**GPO49**	GPIO49_DOUT	**GPIO 49 Data Output Value** 0: Output 0 1: Output 1
0	**GPO48**	GPIO48_DOUT	**GPIO 48 Data Output Value** 0: Output 0 1: Output 1

10005540 **GPIO_DOUT5** **GPIO Data Output Register 5** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO79	GPO78	GPO77	GPO76	GPO75	GPO74	GPO73	GPO72	GPO71	GPO70	GPO69	GPO68	GPO67	GPO66	GPO65	GPO64
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15	**GPO79**	GPIO79_DOUT	**GPIO 79 Data Output Value** 0: Output 0 1: Output 1
14	**GPO78**	GPIO78_DOUT	**GPIO 78 Data Output Value** 0: Output 0 1: Output 1
13	**GPO77**	GPIO77_DOUT	**GPIO 77 Data Output Value** 0: Output 0 1: Output 1
12	**GPO76**	GPIO76_DOUT	**GPIO 76 Data Output Value** 0: Output 0 1: Output 1
11	**GPO75**	GPIO75_DOUT	**GPIO 75 Data Output Value** 0: Output 0 1: Output 1
10	**GPO74**	GPIO74_DOUT	**GPIO 74 Data Output Value** 0: Output 0 1: Output 1
9	**GPO73**	GPIO73_DOUT	**GPIO 73 Data Output Value** 0: Output 0 1: Output 1
8	**GPO72**	GPIO72_DOUT	**GPIO 72 Data Output Value** 0: Output 0 1: Output 1
7	**GPO71**	GPIO71_DOUT	**GPIO 71 Data Output Value** 0: Output 0 1: Output 1
6	**GPO70**	GPIO70_DOUT	**GPIO 70 Data Output Value** 0: Output 0 1: Output 1
5	**GPO69**	GPIO69_DOUT	**GPIO 69 Data Output Value** 0: Output 0 1: Output 1
4	**GPO68**	GPIO68_DOUT	**GPIO 68 Data Output Value** 0: Output 0 1: Output 1
3	**GPO67**	GPIO67_DOUT	**GPIO 67 Data Output Value** 0: Output 0 1: Output 1
2	**GPO66**	GPIO66_DOUT	**GPIO 66 Data Output Value** 0: Output 0 1: Output 1
1	**GPO65**	GPIO65_DOUT	**GPIO 65 Data Output Value** 0: Output 0 1: Output 1
0	**GPO64**	GPIO64_DOUT	**GPIO 64 Data Output Value** 0: Output 0 1: Output 1

10005550 **GPIO_DOUT6** **GPIO Data Output Register 6** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO95	GPO94	GPO93	GPO92	GPO91	GPO90	GPO89	GPO88	GPO87	GPO86	GPO85	GPO84	GPO83	GPO82	GPO81	GPO80
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO95**	GPIO95_DOUT	**GPIO 95 Data Output Value** 0: Output 0 1: Output 1
14	**Reserved**	Reserved	**Reserved**
13	**Reserved**	Reserved	**Reserved**
12	**Reserved**	Reserved	**Reserved**
11	**Reserved**	Reserved	**Reserved**
10	**GPO90**	GPIO90_DOUT	**GPIO 90 Data Output Value** 0: Output 0 1: Output 1
9	**GPO89**	GPIO89_DOUT	**GPIO 89 Data Output Value** 0: Output 0 1: Output 1
8	**GPO88**	GPIO88_DOUT	**GPIO 88 Data Output Value** 0: Output 0 1: Output 1
7	**GPO87**	GPIO87_DOUT	**GPIO 87 Data Output Value** 0: Output 0 1: Output 1
6	**GPO86**	GPIO86_DOUT	**GPIO 86 Data Output Value** 0: Output 0 1: Output 1
5	**GPO85**	GPIO85_DOUT	**GPIO 85 Data Output Value** 0: Output 0 1: Output 1
4	**GPO84**	GPIO84_DOUT	**GPIO 84 Data Output Value** 0: Output 0 1: Output 1
3	**GPO83**	GPIO83_DOUT	**GPIO 83 Data Output Value** 0: Output 0 1: Output 1
2	**GPO82**	GPIO82_DOUT	**GPIO 82 Data Output Value** 0: Output 0 1: Output 1
1	**GPO81**	GPIO81_DOUT	**GPIO 81 Data Output Value** 0: Output 0 1: Output 1
0	**GPO80**	GPIO80_DOUT	**GPIO 80 Data Output Value** 0: Output 0 1: Output 1

10005560 **GPIO_DOUT7** **GPIO Data Output Register 7** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO111	GPO110	GPO109	GPO108	GPO107	GPO106	GPO105	GPO104	GPO103	GPO102	GPO101	GPO100	GPO99	GPO98	GPO97	GPO96
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO111**	GPIO111_DOUT	**GPIO 111 Data Output Value** 0: Output 0 1: Output 1
14	**GPO110**	GPIO110_DOUT	**GPIO 110 Data Output Value** 0: Output 0 1: Output 1
13	**GPO109**	GPIO109_DOUT	**GPIO 109 Data Output Value** 0: Output 0 1: Output 1
12	**GPO108**	GPIO108_DOUT	**GPIO 108 Data Output Value** 0: Output 0 1: Output 1
11	**GPO107**	GPIO107_DOUT	**GPIO 107 Data Output Value** 0: Output 0 1: Output 1
10	**GPO106**	GPIO106_DOUT	**GPIO 106 Data Output Value** 0: Output 0 1: Output 1
9	**GPO105**	GPIO105_DOUT	**GPIO 105 Data Output Value** 0: Output 0 1: Output 1
8	**GPO104**	GPIO104_DOUT	**GPIO 104 Data Output Value** 0: Output 0 1: Output 1
7	**GPO103**	GPIO103_DOUT	**GPIO 103 Data Output Value** 0: Output 0 1: Output 1
6	**GPO102**	GPIO102_DOUT	**GPIO 102 Data Output Value** 0: Output 0 1: Output 1
5	**GPO101**	GPIO101_DOUT	**GPIO 101 Data Output Value** 0: Output 0 1: Output 1
4	**GPO100**	GPIO100_DOUT	**GPIO 100 Data Output Value** 0: Output 0 1: Output 1
3	**GPO99**	GPIO99_DOUT	**GPIO 99 Data Output Value** 0: Output 0 1: Output 1
2	**GPO98**	GPIO98_DOUT	**GPIO 98 Data Output Value** 0: Output 0

Bit(s)	Mnemonic	Name	Description
			1: Output 1
1	**GPO97**	GPIO97_DOUT	**GPIO 97 Data Output Value** 0: Output 0 1: Output 1
0	**GPO96**	GPIO96_DOUT	**GPIO 96 Data Output Value** 0: Output 0 1: Output 1

10005570 **GPIO_DOUT 8** **GPIO Data Output Register 8** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO127	GPO126	GPO125	GPO124	GPO123	GPO122	GPO121	GPO120	GPO119	GPO118	GPO117	GPO116	GPO115	GPO114	GPO113	GPO112
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**Reserved**	Reserved	**Reserved**
14	**GPO126**	GPIO126_DOUT	**GPIO 126 Data Output Value** 0: Output 0 1: Output 1
13	**GPO125**	GPIO125_DOUT	**GPIO 125 Data Output Value** 0: Output 0 1: Output 1
12	**GPO124**	GPIO124_DOUT	**GPIO 124 Data Output Value** 0: Output 0 1: Output 1
11	**GPO123**	GPIO123_DOUT	**GPIO 123 Data Output Value** 0: Output 0 1: Output 1
10	**GPO122**	GPIO122_DOUT	**GPIO 122 Data Output Value** 0: Output 0 1: Output 1
9	**GPO121**	GPIO121_DOUT	**GPIO 121 Data Output Value** 0: Output 0 1: Output 1
8	**GPO120**	GPIO120_DOUT	**GPIO 120 Data Output Value** 0: Output 0 1: Output 1
7	**GPO119**	GPIO119_DOUT	**GPIO 119 Data Output Value** 0: Output 0 1: Output 1
6	**GPO118**	GPIO118_DOUT	**GPIO 118 Data Output Value** 0: Output 0 1: Output 1
5	**GPO117**	GPIO117_DOUT	**GPIO 117 Data Output Value** 0: Output 0

Bit(s)	Mnemonic	Name	Description
			1: Output 1
4	**GPO116**	GPIO116_DOUT	**GPIO 116 Data Output Value** 0: Output 0 1: Output 1
3	**GPO115**	GPIO115_DOUT	**GPIO 115 Data Output Value** 0: Output 0 1: Output 1
2	**GPO114**	GPIO114_DOUT	**GPIO 114 Data Output Value** 0: Output 0 1: Output 1
1	**GPO113**	GPIO113_DOUT	**GPIO 113 Data Output Value** 0: Output 0 1: Output 1
0	**GPO112**	GPIO112_DOUT	**GPIO 112 Data Output Value** 0: Output 0 1: Output 1

10005580 **GPIO_DOUT9** **GPIO Data Output Register 9** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO143	GPO142	GPO141	GPO140	GPO139	GPO138	GPO137	GPO136	GPO135	GPO134	GPO133	GPO132	GPO131	GPO130	GPO129	GPO128
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005590 **GPIO_DOUT10** **GPIO Data Output Register 10** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO159	GPO158	GPO157	GPO156	GPO155	GPO154	GPO153	GPO152	GPO151	GPO150	GPO149	GPO148	GPO147	GPO146	GPO145	GPO144
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100055A0 **GPIO_DOUT11** **GPIO Data Output Register 11** **0000**

100055A0 GPIO_DOUT11 GPIO Data Output Register 11 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO175	GPO174	GPO173	GPO172	GPO171	GPO170	GPO169	GPO168	GPO167	GPO166	GPO165	GPO164	GPO163	GPO162	GPO161	GPO160
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100055B0 GPIO_DOUT12 GPIO Data Output Register 12 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO191	GPO190	GPO189	GPO188	GPO187	GPO186	GPO185	GPO184	GPO183	GPO182	GPO181	GPO180	GPO179	GPO178	GPO177	GPO176
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO191**	GPIO191_DOUT	**GPIO 191 Data Output Value** 0: Output 0 1: Output 1
14	**GPO190**	GPIO190_DOUT	**GPIO 190 Data Output Value** 0: Output 0 1: Output 1
13	**GPO189**	GPIO189_DOUT	**GPIO 189 Data Output Value** 0: Output 0 1: Output 1
12	**GPO188**	GPIO188_DOUT	**GPIO 188 Data Output Value** 0: Output 0 1: Output 1
11:0	**Reserved**	Reserved	**Reserved**

100055C0 GPIO_DOUT13 GPIO Data Output Register 13 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO207	GPO206	GPO205	GPO204	GPO203	GPO202	GPO201	GPO200	GPO199	GPO198	GPO197	GPO196	GPO195	GPO194	GPO193	GPO192
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15	**GPO207**	GPIO207_DOUT	**GPIO 207 Data Output Value** 0: Output 0 1: Output 1
14	**GPO206**	GPIO206_DOUT	**GPIO 206 Data Output Value** 0: Output 0 1: Output 1
13	**GPO205**	GPIO205_DOUT	**GPIO 205 Data Output Value** 0: Output 0 1: Output 1
12	**GPO204**	GPIO204_DOUT	**GPIO 204 Data Output Value** 0: Output 0 1: Output 1
11	**GPO203**	GPIO203_DOUT	**GPIO 203 Data Output Value** 0: Output 0 1: Output 1
10	**GPO202**	GPIO202_DOUT	**GPIO 202 Data Output Value** 0: Output 0 1: Output 1
9	**GPO201**	GPIO201_DOUT	**GPIO 201 Data Output Value** 0: Output 0 1: Output 1
8	**GPO200**	GPIO200_DOUT	**GPIO 200 Data Output Value** 0: Output 0 1: Output 1
7	**GPO199**	GPIO199_DOUT	**GPIO 199 Data Output Value** 0: Output 0 1: Output 1
6	**GPO198**	GPIO198_DOUT	**GPIO 198 Data Output Value** 0: Output 0 1: Output 1
5	**GPO197**	GPIO197_DOUT	**GPIO 197 Data Output Value** 0: Output 0 1: Output 1
4	**GPO196**	GPIO196_DOUT	**GPIO 196 Data Output Value** 0: Output 0 1: Output 1
3	**GPO195**	GPIO195_DOUT	**GPIO 195 Data Output Value** 0: Output 0 1: Output 1
2	**GPO194**	GPIO194_DOUT	**GPIO 194 Data Output Value** 0: Output 0 1: Output 1
1	**GPO193**	GPIO193_DOUT	**GPIO 193 Data Output Value** 0: Output 0 1: Output 1
0	**GPO192**	GPIO192_DOUT	**GPIO 192 Data Output Value** 0: Output 0 1: Output 1

100055D0 **GPIO_DOUT14** **GPIO Data Output Register 14** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO223	GPO222	GPO221	GPO220	GPO219	GPO218	GPO217	GPO216	GPO215	GPO214	GPO213	GPO212	GPO211	GPO210	GPO209	GPO208
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:2	**Reserved**	Reserved	**Reserved**
1	**GPO209**	GPIO209_DOUT	**GPIO 209 Data Output Value** 0: Output 0 1: Output 1
0	**GPO208**	GPIO208_DOUT	**GPIO 208 Data Output Value** 0: Output 0 1: Output 1

100055E0 **GPIO_DOUT15** **GPIO Data Output Register 15** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO239	GPO238	GPO237	GPO236	GPO235	GPO234	GPO233	GPO232	GPO231	GPO230	GPO229	GPO228	GPO227	GPO226	GPO225	GPO224
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO239**	GPIO239_DOUT	**GPIO 239 Data Output Value** 0: Output 0 1: Output 1
14	**GPO238**	GPIO238_DOUT	**GPIO 238 Data Output Value** 0: Output 0 1: Output 1
13	**GPO237**	GPIO237_DOUT	**GPIO 237 Data Output Value** 0: Output 0 1: Output 1
12	**GPO236**	GPIO236_DOUT	**GPIO 236 Data Output Value** 0: Output 0 1: Output 1
11:0	**Reserved**	Reserved	**Reserved**

100055F0 **GPIO_DOUT16** **GPIO Data Output Register 16** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO	GPO

100055F0 **GPIO_DOUT16** **GPIO Data Output Register 16** **0000**

	55	54	53	52	51	50	49	48	47	46	45	44	43	42	41	40
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO255**	GPIO255_DOUT	**GPIO 255 Data Output Value** 0: Output 0 1: Output 1
14	**GPO254**	GPIO254_DOUT	**GPIO 254 Data Output Value** 0: Output 0 1: Output 1
13	**GPO253**	GPIO253_DOUT	**GPIO 253 Data Output Value** 0: Output 0 1: Output 1
12	**GPO252**	GPIO252_DOUT	**GPIO 252 Data Output Value** 0: Output 0 1: Output 1
11	**GPO251**	GPIO251_DOUT	**GPIO 251 Data Output Value** 0: Output 0 1: Output 1
10	**GPO250**	GPIO250_DOUT	**GPIO 250 Data Output Value** 0: Output 0 1: Output 1
9	**GPO249**	GPIO249_DOUT	**GPIO 249 Data Output Value** 0: Output 0 1: Output 1
8	**GPO248**	GPIO248_DOUT	**GPIO 248 Data Output Value** 0: Output 0 1: Output 1
7	**GPO247**	GPIO247_DOUT	**GPIO 247 Data Output Value** 0: Output 0 1: Output 1
6	**GPO246**	GPIO246_DOUT	**GPIO 246 Data Output Value** 0: Output 0 1: Output 1
5	**GPO245**	GPIO245_DOUT	**GPIO 245 Data Output Value** 0: Output 0 1: Output 1
4	**GPO244**	GPIO244_DOUT	**GPIO 244 Data Output Value** 0: Output 0 1: Output 1
3	**GPO243**	GPIO243_DOUT	**GPIO 243 Data Output Value** 0: Output 0 1: Output 1
2	**GPO242**	GPIO242_DOUT	**GPIO 242 Data Output Value** 0: Output 0 1: Output 1
1	**GPO241**	GPIO241_DOUT	**GPIO 241 Data Output Value** 0: Output 0

Bit(s)	Mnemonic	Name	Description
			1: Output 1
0	**GPO240**	GPIO240_DOUT	**GPIO 240 Data Output Value** 0: Output 0 1: Output 1

10005600 **GPIO_DOUT17** **GPIO Data Output Register 17** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPO271	GPO270	GPO269	GPO268	GPO267	GPO266	GPO265	GPO264	GPO263	GPO262	GPO261	GPO260	GPO259	GPO258	GPO257	GPO256
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**GPO271**	GPIO271_DOUT	**GPIO 271 Data Output Value** 0: Output 0 1: Output 1
14	**GPO270**	GPIO270_DOUT	**GPIO 270 Data Output Value** 0: Output 0 1: Output 1
13	**GPO269**	GPIO269_DOUT	**GPIO 269 Data Output Value** 0: Output 0 1: Output 1
12	**GPO268**	GPIO268_DOUT	**GPIO 268 Data Output Value** 0: Output 0 1: Output 1
11	**GPO267**	GPIO267_DOUT	**GPIO 267 Data Output Value** 0: Output 0 1: Output 1
10	**GPO266**	GPIO266_DOUT	**GPIO 266 Data Output Value** 0: Output 0 1: Output 1
9	**GPO265**	GPIO265_DOUT	**GPIO 265 Data Output Value** 0: Output 0 1: Output 1
8	**GPO264**	GPIO264_DOUT	**GPIO 264 Data Output Value** 0: Output 0 1: Output 1
7	**GPO263**	GPIO263_DOUT	**GPIO 263 Data Output Value** 0: Output 0 1: Output 1
6	**GPO262**	GPIO262_DOUT	**GPIO 262 Data Output Value** 0: Output 0 1: Output 1
5	**GPO261**	GPIO261_DOUT	**GPIO 261 Data Output Value** 0: Output 0 1: Output 1

Bit(s)	Mnemonic	Name	Description
4	**GPO260**	GPIO260_DOUT	**GPIO 260 Data Output Value** 0: Output 0 1: Output 1
3	**GPO259**	GPIO259_DOUT	**GPIO 259 Data Output Value** 0: Output 0 1: Output 1
2	**GPO258**	GPIO258_DOUT	**GPIO 258 Data Output Value** 0: Output 0 1: Output 1
1	**GPO257**	GPIO257_DOUT	**GPIO 257 Data Output Value** 0: Output 0 1: Output 1
0	**GPO256**	GPIO256_DOUT	**GPIO 256 Data Output Value** 0: Output 0 1: Output 1

10005610 **GPIO_DOUT18** **GPIO Data Output Register 18** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**GPO279**	**GPO278**	**GPO277**	**GPO276**	**GPO275**	**GPO274**	**GPO273**	**GPO272**
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	**Reserved**	Reserved	**Reserved**
6	**GPO278**	GPIO278_DOUT	**GPIO 278 Data Output Value** 0: Output 0 1: Output 1
5	**Reserved**	Reserved	**Reserved**
4	**GPO276**	GPIO276_DOUT	**GPIO 276 Data Output Value** 0: Output 0 1: Output 1
3	**GPO275**	GPIO275_DOUT	**GPIO 275 Data Output Value** 0: Output 0 1: Output 1
2	**GPO274**	GPIO274_DOUT	**GPIO 274 Data Output Value** 0: Output 0 1: Output 1
1	**Reserved**	Reserved	**Reserved**
0	**GPO272**	GPIO272_DOUT	**GPIO 272 Data Output Value** 0: Output 0 1: Output 1

10005620 MSDC3_CTRL6 MSDC 3 Pad Control Register 00AC

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3PAD_BACKUP						MS3PAD_RDSEL						MS3PAD_TDSEL			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0

Bit(s)	Mnemonic	Name	Description
15:10		MS3PAD_BACKUP	Reserved
9:4		MS3PAD_RDSEL	RDSEL control register for MS3 PAD
3:0		MS3PAD_TDSEL	TDSEL control register for MS3 PAD

10005630 GPIO_DIN1 GPIO Data Input Register 1 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI15	GPI14	GPI13	GPI12	GPI11	GPI10	GPI9	GPI8	GPI7	GPI6	GPI5	GPI4	GPI3	GPI2	GPI1	GPI0
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	GPI15	GPIO15_DIN	GPIO 15 Data Input Value
14	GPI14	GPIO14_DIN	GPIO 14 Data Input Value
13	GPI13	GPIO13_DIN	GPIO 13 Data Input Value
12	GPI12	GPIO12_DIN	GPIO 12 Data Input Value
11	GPI11	GPIO11_DIN	GPIO 11 Data Input Value
10	GPI10	GPIO10_DIN	GPIO 10 Data Input Value
9	GPI9	GPIO9_DIN	GPIO 9 Data Input Value
8	GPI8	GPIO8_DIN	GPIO 8 Data Input Value
7	GPI7	GPIO7_DIN	GPIO 7 Data Input Value
6	GPI6	GPIO6_DIN	GPIO 6 Data Input Value
5	GPI5	GPIO5_DIN	GPIO 5 Data Input Value
4	GPI4	GPIO4_DIN	GPIO 4 Data Input Value
3	GPI3	GPIO3_DIN	GPIO 3 Data Input Value
2	GPI2	GPIO2_DIN	GPIO 2 Data Input Value
1	GPI1	GPIO1_DIN	GPIO 1 Data Input Value
0	GPI0	GPIO0_DIN	GPIO 0 Data Input Value

10005640 GPIO_DIN2 GPIO Data Input Register 2 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI31	GPI30	GPI29	GPI28	GPI27	GPI26	GPI25	GPI24	GPI23	GPI22	GPI21	GPI20	GPI19	GPI18	GPI17	GPI16
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

10005640 GPIO_DIN2 GPIO Data Input Register 2 NA

Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI31**	GPIO31_DIN	**GPIO 31 Data Input Value**
14	**GPI30**	GPIO30_DIN	**GPIO 30 Data Input Value**
13	**GPI29**	GPIO29_DIN	**GPIO 29 Data Input Value**
12	**GPI28**	GPIO28_DIN	**GPIO 28 Data Input Value**
11	**GPI27**	GPIO27_DIN	**GPIO 27 Data Input Value**
10	**GPI26**	GPIO26_DIN	**GPIO 26 Data Input Value**
9	**GPI25**	GPIO25_DIN	**GPIO 25 Data Input Value**
8	**GPI24**	GPIO24_DIN	**GPIO 24 Data Input Value**
7	**GPI23**	GPIO23_DIN	**GPIO 23 Data Input Value**
6	**GPI22**	GPIO22_DIN	**GPIO 22 Data Input Value**
5	**GPI21**	GPIO21_DIN	**GPIO 21 Data Input Value**
4	**GPI20**	GPIO20_DIN	**GPIO 20 Data Input Value**
3	**GPI19**	GPIO19_DIN	**GPIO 19 Data Input Value**
2	**GPI18**	GPIO18_DIN	**GPIO 18 Data Input Value**
1	**GPI17**	GPIO17_DIN	**GPIO 17 Data Input Value**
0	**GPI16**	GPIO16_DIN	**GPIO 16 Data Input Value**

10005650 GPIO_DIN3 GPIO Data Input Register 3 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI47	GPI46	GPI45	GPI44	GPI43	GPI42	GPI41	GPI40	GPI39	GPI38	GPI37	GPI36	GPI35	GPI34	GPI33	GPI32
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI47**	GPIO47_DIN	**GPIO 47 Data Input Value**
14	**GPI46**	GPIO46_DIN	**GPIO 46 Data Input Value**
13	**GPI45**	GPIO45_DIN	**GPIO 45 Data Input Value**
12	**GPI44**	GPIO44_DIN	**GPIO 44 Data Input Value**
11	**GPI43**	GPIO43_DIN	**GPIO 43 Data Input Value**
10	**GPI42**	GPIO42_DIN	**GPIO 42 Data Input Value**
9	**GPI41**	GPIO41_DIN	**GPIO 41 Data Input Value**
8	**GPI40**	GPIO40_DIN	**GPIO 40 Data Input Value**
7	**GPI39**	GPIO39_DIN	**GPIO 39 Data Input Value**
6	**GPI38**	GPIO38_DIN	**GPIO 38 Data Input Value**
5	**GPI37**	GPIO37_DIN	**GPIO 37 Data Input Value**
4	**GPI36**	GPIO36_DIN	**GPIO 36 Data Input Value**
3	**GPI35**	GPIO35_DIN	**GPIO 35 Data Input Value**
2	**GPI34**	GPIO34_DIN	**GPIO 34 Data Input Value**

Bit(s)	Mnemonic	Name	Description
1	**GPI33**	GPIO33_DIN	**GPIO 33 Data Input Value**
0	**GPI32**	GPIO32_DIN	**GPIO 32 Data Input Value**

10005660 GPIO_DIN4 GPIO Data Input Register 4 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI63	GPI62	GPI61	GPI60	GPI59	GPI58	GPI57	GPI56	GPI55	GPI54	GPI53	GPI52	GPI51	GPI50	GPI49	GPI48
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI63**	GPIO63_DIN	**GPIO 63 Data Input Value**
14	**GPI62**	GPIO62_DIN	**GPIO 62 Data Input Value**
13	**GPI61**	GPIO61_DIN	**GPIO 61 Data Input Value**
12	**GPI60**	GPIO60_DIN	**GPIO 60 Data Input Value**
11	**Reserved**	Reserved	**Reserved**
10	**GPI58**	GPIO58_DIN	**GPIO 58 Data Input Value**
9	**GPI57**	GPIO57_DIN	**GPIO 57 Data Input Value**
8	**GPI56**	GPIO56_DIN	**GPIO 56 Data Input Value**
7	**GPI55**	GPIO55_DIN	**GPIO 55 Data Input Value**
6	**GPI54**	GPIO54_DIN	**GPIO 54 Data Input Value**
5	**GPI53**	GPIO53_DIN	**GPIO 53 Data Input Value**
4	**GPI52**	GPIO52_DIN	**GPIO 52 Data Input Value**
3	**GPI51**	GPIO51_DIN	**GPIO 51 Data Input Value**
2	**GPI50**	GPIO50_DIN	**GPIO 50 Data Input Value**
1	**GPI49**	GPIO49_DIN	**GPIO 49 Data Input Value**
0	**GPI48**	GPIO48_DIN	**GPIO 48 Data Input Value**

10005670 GPIO_DIN5 GPIO Data Input Register 5 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI79	GPI78	GPI77	GPI76	GPI75	GPI74	GPI73	GPI72	GPI71	GPI70	GPI69	GPI68	GPI67	GPI66	GPI65	GPI64
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI79**	GPIO79_DIN	**GPIO 79 Data Input Value**
14	**GPI78**	GPIO78_DIN	**GPIO 78 Data Input Value**
13	**GPI77**	GPIO77_DIN	**GPIO 77 Data Input Value**
12	**GPI76**	GPIO76_DIN	**GPIO 76 Data Input Value**
11	**GPI75**	GPIO75_DIN	**GPIO 75 Data Input Value**

Bit(s)	Mnemonic	Name	Description
10	**GPI74**	GPIO74_DIN	**GPIO 74 Data Input Value**
9	**GPI73**	GPIO73_DIN	**GPIO 73 Data Input Value**
8	**GPI72**	GPIO72_DIN	**GPIO 72 Data Input Value**
7	**GPI71**	GPIO71_DIN	**GPIO 71 Data Input Value**
6	**GPI70**	GPIO70_DIN	**GPIO 70 Data Input Value**
5	**GPI69**	GPIO69_DIN	**GPIO 69 Data Input Value**
4	**GPI68**	GPIO68_DIN	**GPIO 68 Data Input Value**
3	**GPI67**	GPIO67_DIN	**GPIO 67 Data Input Value**
2	**GPI66**	GPIO66_DIN	**GPIO 66 Data Input Value**
1	**GPI65**	GPIO65_DIN	**GPIO 65 Data Input Value**
0	**GPI64**	GPIO64_DIN	**GPIO 64 Data Input Value**

10005680 GPIO_DIN6 GPIO Data Input Register 6 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI95	GPI94	GPI93	GPI92	GPI91	GPI90	GPI89	GPI88	GPI87	GPI86	GPI85	GPI84	GPI83	GPI82	GPI81	GPI80
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI95**	GPIO95_DIN	**GPIO 95 Data Input Value**
14	**Reserved**	Reserved	**Reserved**
13	**Reserved**	Reserved	**Reserved**
12	**Reserved**	Reserved	**Reserved**
11	**Reserved**	Reserved	**Reserved**
10	**GPI90**	GPIO90_DIN	**GPIO 90 Data Input Value**
9	**GPI89**	GPIO89_DIN	**GPIO 89 Data Input Value**
8	**GPI88**	GPIO88_DIN	**GPIO 88 Data Input Value**
7	**GPI87**	GPIO87_DIN	**GPIO 87 Data Input Value**
6	**GPI86**	GPIO86_DIN	**GPIO 86 Data Input Value**
5	**GPI85**	GPIO85_DIN	**GPIO 85 Data Input Value**
4	**GPI84**	GPIO84_DIN	**GPIO 84 Data Input Value**
3	**GPI83**	GPIO83_DIN	**GPIO 83 Data Input Value**
2	**GPI82**	GPIO82_DIN	**GPIO 82 Data Input Value**
1	**GPI81**	GPIO81_DIN	**GPIO 81 Data Input Value**
0	**GPI80**	GPIO80_DIN	**GPIO 80 Data Input Value**

10005690 GPIO_DIN7 GPIO Data Input Register 7 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI111	GPI110	GPI109	GPI108	GPI107	GPI106	GPI105	GPI104	GPI103	GPI102	GPI101	GPI100	GPI99	GPI98	GPI97	GPI96
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

10005690 GPIO_DIN7 GPIO Data Input Register 7 NA

Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI111**	GPIO111_DIN	**GPIO 111 Data Input Value**
14	**GPI110**	GPIO110_DIN	**GPIO 110 Data Input Value**
13	**GPI109**	GPIO109_DIN	**GPIO 109 Data Input Value**
12	**GPI108**	GPIO108_DIN	**GPIO 108 Data Input Value**
11	**GPI107**	GPIO107_DIN	**GPIO 107 Data Input Value**
10	**GPI106**	GPIO106_DIN	**GPIO 106 Data Input Value**
9	**GPI105**	GPIO105_DIN	**GPIO 105 Data Input Value**
8	**GPI104**	GPIO104_DIN	**GPIO 104 Data Input Value**
7	**GPI103**	GPIO103_DIN	**GPIO 103 Data Input Value**
6	**GPI102**	GPIO102_DIN	**GPIO 102 Data Input Value**
5	**GPI101**	GPIO101_DIN	**GPIO 101 Data Input Value**
4	**GPI100**	GPIO100_DIN	**GPIO 100 Data Input Value**
3	**GPI99**	GPIO99_DIN	**GPIO 99 Data Input Value**
2	**GPI98**	GPIO98_DIN	**GPIO 98 Data Input Value**
1	**GPI97**	GPIO97_DIN	**GPIO 97 Data Input Value**
0	**GPI96**	GPIO96_DIN	**GPIO 96 Data Input Value**

100056A0 GPIO_DIN8 GPIO Data Input Register 8 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**GPI127**	**GPI126**	**GPI125**	**GPI124**	**GPI123**	**GPI122**	**GPI121**	**GPI120**	**GPI119**	**GPI118**	**GPI117**	**GPI116**	**GPI115**	**GPI114**	**GPI113**	**GPI112**
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**Reserved**	Reserved	**Reserved**
14	**GPI126**	GPIO126_DIN	**GPIO 126 Data Input Value**
13	**GPI125**	GPIO125_DIN	**GPIO 125 Data Input Value**
12	**GPI124**	GPIO124_DIN	**GPIO 124 Data Input Value**
11	**GPI123**	GPIO123_DIN	**GPIO 123 Data Input Value**
10	**GPI122**	GPIO122_DIN	**GPIO 122 Data Input Value**
9	**GPI121**	GPIO121_DIN	**GPIO 121 Data Input Value**
8	**GPI120**	GPIO120_DIN	**GPIO 120 Data Input Value**
7	**GPI119**	GPIO119_DIN	**GPIO 119 Data Input Value**
6	**GPI118**	GPIO118_DIN	**GPIO 118 Data Input Value**
5	**GPI117**	GPIO117_DIN	**GPIO 117 Data Input Value**
4	**GPI116**	GPIO116_DIN	**GPIO 116 Data Input Value**
3	**GPI115**	GPIO115_DIN	**GPIO 115 Data Input Value**

Bit(s)	Mnemonic	Name	Description
2	**GPI114**	GPIO114_DIN	**GPIO 114 Data Input Value**
1	**GPI113**	GPIO113_DIN	**GPIO 113 Data Input Value**
0	**GPI112**	GPIO112_DIN	**GPIO 112 Data Input Value**

100056B0 **GPIO_DIN9** **GPIO Data Input Register 9** **NA**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI143	GPI142	GPI141	GPI140	GPI139	GPI138	GPI137	GPI136	GPI135	GPI134	GPI133	GPI132	GPI131	GPI130	GPI129	GPI128
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100056C0 **GPIO_DIN10** **GPIO Data Input Register 10** **NA**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI159	GPI158	GPI157	GPI156	GPI155	GPI154	GPI153	GPI152	GPI151	GPI150	GPI149	GPI148	GPI147	GPI146	GPI145	GPI144
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100056D0 **GPIO_DIN11** **GPIO Data Input Register 11** **NA**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI175	GPI174	GPI173	GPI172	GPI171	GPI170	GPI169	GPI168	GPI167	GPI166	GPI165	GPI164	GPI163	GPI162	GPI161	GPI160
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

100056E0 **GPIO_DIN12** **GPIO Data Input Register 12** **NA**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI	GPI

100056E0 GPIO_DIN12 GPIO Data Input Register 12 NA

e	191	190	189	188	187	186	185	184	183	182	181	180	179	178	177	176
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI191**	GPIO191_DIN	**GPIO 191 Data Input Value**
14	**GPI190**	GPIO190_DIN	**GPIO 190 Data Input Value**
13	**GPI189**	GPIO189_DIN	**GPIO 189 Data Input Value**
12	**GPI188**	GPIO188_DIN	**GPIO 188 Data Input Value**
11:0	**Reserved**	Reserved	**Reserved**

100056F0 GPIO_DIN13 GPIO Data Input Register 13 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI207	GPI206	GPI205	GPI204	GPI203	GPI202	GPI201	GPI200	GPI199	GPI198	GPI197	GPI196	GPI195	GPI194	GPI193	GPI192
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI207**	GPIO207_DIN	**GPIO 207 Data Input Value**
14	**GPI206**	GPIO206_DIN	**GPIO 206 Data Input Value**
13	**GPI205**	GPIO205_DIN	**GPIO 205 Data Input Value**
12	**GPI204**	GPIO204_DIN	**GPIO 204 Data Input Value**
11	**GPI203**	GPIO203_DIN	**GPIO 203 Data Input Value**
10	**GPI202**	GPIO202_DIN	**GPIO 202 Data Input Value**
9	**GPI201**	GPIO201_DIN	**GPIO 201 Data Input Value**
8	**GPI200**	GPIO200_DIN	**GPIO 200 Data Input Value**
7	**GPI199**	GPIO199_DIN	**GPIO 199 Data Input Value**
6	**GPI198**	GPIO198_DIN	**GPIO 198 Data Input Value**
5	**GPI197**	GPIO197_DIN	**GPIO 197 Data Input Value**
4	**GPI196**	GPIO196_DIN	**GPIO 196 Data Input Value**
3	**GPI195**	GPIO195_DIN	**GPIO 195 Data Input Value**
2	**GPI194**	GPIO194_DIN	**GPIO 194 Data Input Value**
1	**GPI193**	GPIO193_DIN	**GPIO 193 Data Input Value**
0	**GPI192**	GPIO192_DIN	**GPIO 192 Data Input Value**

10005700 GPIO_DIN14 GPIO Data Input Register 14 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI223	GPI222	GPI221	GPI220	GPI219	GPI218	GPI217	GPI216	GPI215	GPI214	GPI213	GPI212	GPI211	GPI210	GPI209	GPI208
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

10005700 GPIO_DIN14 GPIO Data Input Register 14 NA

Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15:2	**Reserved**	Reserved	**Reserved**
1	**GPI209**	GPIO209_DIN	**GPIO 209 Data Input Value**
0	**GPI208**	GPIO208_DIN	**GPIO 208 Data Input Value**

10005710 GPIO_DIN15 GPIO Data Input Register 15 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI239	GPI238	GPI237	GPI236	GPI235	GPI234	GPI233	GPI232	GPI231	GPI230	GPI229	GPI228	GPI227	GPI226	GPI225	GPI224
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI239**	GPIO239_DIN	**GPIO 239 Data Input Value**
14	**GPI238**	GPIO238_DIN	**GPIO 238 Data Input Value**
13	**GPI237**	GPIO237_DIN	**GPIO 237 Data Input Value**
12	**GPI236**	GPIO236_DIN	**GPIO 236 Data Input Value**
11:0	**Reserved**	Reserved	**Reserved**

10005720 GPIO_DIN16 GPIO Data Input Register 16 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI255	GPI254	GPI253	GPI252	GPI251	GPI250	GPI249	GPI248	GPI247	GPI246	GPI245	GPI244	GPI243	GPI242	GPI241	GPI240
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI255**	GPIO255_DIN	**GPIO 255 Data Input Value**
14	**GPI254**	GPIO254_DIN	**GPIO 254 Data Input Value**
13	**GPI253**	GPIO253_DIN	**GPIO 253 Data Input Value**
12	**GPI252**	GPIO252_DIN	**GPIO 252 Data Input Value**
11	**GPI251**	GPIO251_DIN	**GPIO 251 Data Input Value**
10	**GPI250**	GPIO250_DIN	**GPIO 250 Data Input Value**
9	**GPI249**	GPIO249_DIN	**GPIO 249 Data Input Value**
8	**GPI248**	GPIO248_DIN	**GPIO 248 Data Input Value**
7	**GPI247**	GPIO247_DIN	**GPIO 247 Data Input Value**
6	**GPI246**	GPIO246_DIN	**GPIO 246 Data Input Value**
5	**GPI245**	GPIO245_DIN	**GPIO 245 Data Input Value**
4	**GPI244**	GPIO244_DIN	**GPIO 244 Data Input Value**

Bit(s)	Mnemonic	Name	Description
3	**GPI243**	GPIO243_DIN	**GPIO 243 Data Input Value**
2	**GPI242**	GPIO242_DIN	**GPIO 242 Data Input Value**
1	**GPI241**	GPIO241_DIN	**GPIO 241 Data Input Value**
0	**GPI240**	GPIO240_DIN	**GPIO 240 Data Input Value**

10005730 GPIO_DIN17 GPIO Data Input Register 17 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPI271	GPI270	GPI269	GPI268	GPI267	GPI266	GPI265	GPI264	GPI263	GPI262	GPI261	GPI260	GPI259	GPI258	GPI257	GPI256
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
15	**GPI271**	GPIO271_DIN	**GPIO 271 Data Input Value**
14	**GPI270**	GPIO270_DIN	**GPIO 270 Data Input Value**
13	**GPI269**	GPIO269_DIN	**GPIO 269 Data Input Value**
12	**GPI268**	GPIO268_DIN	**GPIO 268 Data Input Value**
11	**GPI267**	GPIO267_DIN	**GPIO 267 Data Input Value**
10	**GPI266**	GPIO266_DIN	**GPIO 266 Data Input Value**
9	**GPI265**	GPIO265_DIN	**GPIO 265 Data Input Value**
8	**GPI264**	GPIO264_DIN	**GPIO 264 Data Input Value**
7	**GPI263**	GPIO263_DIN	**GPIO 263 Data Input Value**
6	**GPI262**	GPIO262_DIN	**GPIO 262 Data Input Value**
5	**GPI261**	GPIO261_DIN	**GPIO 261 Data Input Value**
4	**GPI260**	GPIO260_DIN	**GPIO 260 Data Input Value**
3	**GPI259**	GPIO259_DIN	**GPIO 259 Data Input Value**
2	**GPI258**	GPIO258_DIN	**GPIO 258 Data Input Value**
1	**GPI257**	GPIO257_DIN	**GPIO 257 Data Input Value**
0	**GPI256**	GPIO256_DIN	**GPIO 256 Data Input Value**

10005740 GPIO_DIN18 GPIO Data Input Register 18 NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									GPI279	GPI278	GPI277	GPI276	GPI275	GPI274	GPI273	GPI272
Type									RO	RO	RO	RO	RO	RO	RO	RO
Reset									X	X	X	X	X	X	X	X

Bit(s)	Mnemonic	Name	Description
7	**Reserved**	Reserved	**Reserved**
6	**GPI278**	GPIO278_DIN	**GPIO 278 Data Input Value**

Bit(s)	Mnemonic	Name	Description
5	**Reserved**	Reserved	**Reserved**
4	**GPI276**	GPIO276_DIN	**GPIO 276 Data Input Value**
3	**GPI275**	GPIO275_DIN	**GPIO 275 Data Input Value**
2	**GPI274**	GPIO274_DIN	**GPIO 274 Data Input Value**
1	**Reserved**	Reserved	**Reserved**
0	**GPI272**	GPIO272_DIN	**GPIO 272 Data Input Value**

10005760 GPIO_MODE1 GPIO Mode Control Register 1 1209

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO4_M			GPIO3_M			GPIO2_M			GPIO1_M			GPIO0_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO4_M**	GPIO4_MODE	**GPIO 4 Mode Selection** 0: GPIO4 (IO) 1: PWRAP_SPICS_B_I (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO3_M**	GPIO3_MODE	**GPIO 3 Mode Selection** 0: GPIO3 (IO) 1: PWRAP_SPICK_I (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO2_M**	GPIO2_MODE	**GPIO 2 Mode Selection** 0: GPIO2 (IO) 1: PWRAP_INT (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO1_M**	GPIO1_MODE	**GPIO 1 Mode Selection** 0: GPIO1 (IO) 1: PWRAP_SPIDI (IO) 2: PWRAP_SPIDO (IO) 3: Reserved 4: Reserved 5: Reserved 6: Reserved

Bit(s)	Mnemonic	Name	Description
			7: Reserved
2:0	GPIO0_M	GPIO0_MODE	**GPIO 0 Mode Selection** 0: GPIO0 (IO) 1: PWRAP_SPIDO (IO) 2: PWRAP_SPIDI (IO) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005770 **GPIO_MODE2** **GPIO Mode Control Register 2** **0009**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO9_M			GPIO8_M			GPIO7_M			GPIO6_M			GPIO5_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	GPIO9_M	GPIO9_MODE	**GPIO 9 Mode Selection** 0: GPIO9 (IO) 1: SPI1_MO (O) 2: SPI1_MI (I) 3: EXT_FRAME_SYNC (I) 4: KCOL2 (IO) 5: Reserved 6: Reserved 7: DBG_MON_B[14] (IO)
11:9	GPIO8_M	GPIO8_MODE	**GPIO 8 Mode Selection** 0: GPIO8 (IO) 1: SPI1_MI (I) 2: SPI1_MO (O) 3: Reserved 4: KCOL1 (IO) 5: Reserved 6: Reserved 7: DBG_MON_B[13] (IO)
8:6	GPIO7_M	GPIO7_MODE	**GPIO 7 Mode Selection** 0: GPIO7 (IO) 1: SPI1_CS (O) 2: Reserved 3: Reserved 4: KCOL0 (IO) 5: Reserved 6: Reserved 7: DBG_MON_B[12] (IO)
5:3	GPIO6_M	GPIO6_MODE	**GPIO 6 Mode Selection** 0: GPIO6 (IO) 1: PWRAP_SPICS2_B_I (O) 2: Reserved 3: Reserved 4: Reserved

Bit(s)	Mnemonic	Name	Description
			5: ANT_SEL0 (O) 6: Reserved 7: DBG_MON_A[0] (IO)
2:0	**GPIO5_M**	GPIO5_MODE	**GPIO 5 Mode Selection** 0: GPIO5 (IO) 1: PWRAP_SPICK2_I (O) 2: Reserved 3: Reserved 4: Reserved 5: ANT_SEL1 (O) 6: Reserved 7: DBG_MON_A[1] (IO)

10005780 **GPIO_MODE3** **GPIO Mode Control Register 3** **0249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO14_M			GPIO13_M			GPIO12_M			GPIO11_M			GPIO10_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO14_M**	GPIO14_MODE	**GPIO 14 Mode Selection** 0: GPIO14 (IO) 1: URXD2 (I) 2: UTXD2 (O) 3: Reserved 4: Reserved 5: SRCCLKENAI2 (I) 6: Reserved 7: DBG_MON_B[30] (IO)
11:9	**GPIO13_M**	GPIO13_MODE	**GPIO 13 Mode Selection** 0: GPIO13 (IO) 1: SRCLKENAI (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO12_M**	GPIO12_MODE	**GPIO 12 Mode Selection** 0: GPIO12 (IO) 1: SRCLKENA (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO11_M**	GPIO11_MODE	**GPIO 11 Mode Selection** 0: GPIO11 (IO) 1: WATCHDOG (O) 2: Reserved

Bit(s)	Mnemonic	Name	Description
			3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	GPIO10_M	GPIO10_MODE	**GPIO 10 Mode Selection** 0: GPIO10 (IO) 1: RTC32K_CK (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005790 **GPIO_MODE4** **GPIO Mode Control Register 4** **0048**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO19_M			GPIO18_M			GPIO17_M			GPIO16_M			GPIO15_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	1	0	0	1	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	GPIO19_M	GPIO19_MODE	**GPIO 19 Mode Selection** 0: GPIO19 (IO) 1: PCM_SYNC (IO) 2: MRG_SYNC (IO) 3: Reserved 4: Reserved 5: CONN_DSP_JINTP (O) 6: WCN_PCM_SYNC (IO) 7: DBG_MON_A[5] (IO)
11:9	GPIO18_M	GPIO18_MODE	**GPIO 18 Mode Selection** 0: GPIO18 (IO) 1: PCM_CLK0 (IO) 2: MRG_CLK (O) 3: Reserved 4: MM_TEST_CK (I) 5: CONN_DSP_JCK (I) 6: WCN_PCM_CLK0 (IO) 7: DBG_MON_A[3] (IO)
8:6	GPIO17_M	GPIO17_MODE	**GPIO 17 Mode Selection** 0: GPIO17 (IO) 1: Reserved 2: Reserved 3: PCM_CLK0 (IO) 4: ANT_SEL2 (O) 5: Reserved 6: Reserved 7: Reserved
5:3	GPIO16_M	GPIO16_MODE	**GPIO 16 Mode Selection** 0: GPIO16 (IO)

Bit(s)	Mnemonic	Name	Description
			1: Reserved 2: Reserved 3: PCM_RX (I) 4: ANT_SEL4 (O) 5: Reserved 6: Reserved 7: Reserved
2:0	GPIO15_M	GPIO15_MODE	**GPIO 15 Mode Selection** 0: GPIO15 (IO) 1: UTXD2 (O) 2: URXD2 (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[31] (IO)

100057A0 **GPIO_MODE5** **GPIO Mode Control Register 5** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO24_M			GPIO23_M			GPIO22_M			GPIO21_M			GPIO20_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	GPIO24_M	GPIO24_MODE	**GPIO 24 Mode Selection** 0: GPIO24 (IO) 1: UCTS1 (I) 2: Reserved 3: KCOL1 (IO) 4: CONN_MCU_DBGACK_N (O) 5: Reserved 6: Reserved 7: DBG_MON_A[28] (IO)
11:9	GPIO23_M	GPIO23_MODE	**GPIO 23 Mode Selection** 0: GPIO23 (IO) 1: URTS0 (O) 2: Reserved 3: KCOL2 (IO) 4: CONN_MCU_TDO (O) 5: EXT_FRAME_SYNC (I) 6: Reserved 7: DBG_MON_A[29] (IO)
8:6	GPIO22_M	GPIO22_MODE	**GPIO 22 Mode Selection** 0: GPIO22 (IO) 1: UCTS0 (I) 2: Reserved 3: KCOL3 (IO) 4: CONN_DSP_JDO (O) 5: EXT_FRAME_SYNC (I) 6: Reserved 7: DBG_MON_A[30] (IO)

Bit(s)	Mnemonic	Name	Description
5:3	**GPIO21_M**	GPIO21_MODE	**GPIO 21 Mode Selection** 0: GPIO21 (IO) 1: PCM_TX (O) 2: MRG_TX (O) 3: MRG_RX (I) 4: PCM_RX (I) 5: CONN_DSP_JMS (I) 6: WCN_PCM_TX (O) 7: DBG_MON_A[2] (IO)
2:0	**GPIO20_M**	GPIO20_MODE	**GPIO 20 Mode Selection** 0: GPIO20 (IO) 1: PCM_RX (I) 2: MRG_RX (I) 3: MRG_TX (O) 4: PCM_TX (O) 5: CONN_DSP_JDI (I) 6: WCN_PCM_RX (I) 7: DBG_MON_A[4] (IO)

100057B0 **GPIO_MODE6** **GPIO Mode Control Register 6** **1000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO29_M			GPIO28_M			GPIO27_M			GPIO26_M			GPIO25_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO29_M**	GPIO29_MODE	**GPIO 29 Mode Selection** 0: GPIO29 (IO) 1: IDDIG (I) 2: MSDC1_WP (I) 3: KROW0 (IO) 4: CONN_MCU_TMS (I) 5: CONN_MCU_AICE_JMSC (IO) 6: Reserved 7: DBG_MON_A[23] (IO)
11:9	**GPIO28_M**	GPIO28_MODE	**GPIO 28 Mode Selection** 0: GPIO28 (IO) 1: DRV_VBUS (O) 2: Reserved 3: KROW1 (IO) 4: CONN_MCU_TRST_B (I) 5: Reserved 6: Reserved 7: DBG_MON_A[24] (IO)
8:6	**GPIO27_M**	GPIO27_MODE	**GPIO 27 Mode Selection** 0: GPIO27 (IO) 1: URTS3 (O) 2: IDDIG_P1 (I) 3: KROW2 (IO) 4: CONN_MCU_TDI (I) 5: Reserved

Bit(s)	Mnemonic	Name	Description
			6: Reserved 7: DBG_MON_A[25] (IO)
5:3	**GPIO26_M**	GPIO26_MODE	**GPIO 26 Mode Selection** 0: GPIO26 (IO) 1: UCTS3 (I) 2: DRV_VBUS_P1 (O) 3: KROW3 (IO) 4: CONN_MCU_TCK0 (I) 5: CONN_MCU_AICE_JCKC (I) 6: Reserved 7: DBG_MON_A[26] (IO)
2:0	**GPIO25_M**	GPIO25_MODE	**GPIO 25 Mode Selection** 0: GPIO25 (IO) 1: URTS1 (O) 2: Reserved 3: KCOL0 (IO) 4: CONN_MCU_DBGI_N (I) 5: Reserved 6: Reserved 7: DBG_MON_A[27] (IO)

100057C0 **GPIO_MODE7** **GPIO Mode Control Register 7** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO34_M			GPIO33_M			GPIO32_M			GPIO31_M			GPIO30_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO34_M**	GPIO34_MODE	**GPIO 34 Mode Selection** 0: GPIO34 (IO) 1: I2S1_DATA_IN (IO) 2: Reserved 3: PCM_RX (I) 4: VDEC_TEST_CK (I) 5: Reserved 6: WCN_PCM_RX (I) 7: DBG_MON_B[7] (IO)
11:9	**GPIO33_M**	GPIO33_MODE	**GPIO 33 Mode Selection** 0: GPIO33 (IO) 1: I2S1_DATA (IO) 2: I2S1_DATA_BYPS (O) 3: PCM_TX (O) 4: IMG_TEST_CK (I) 5: Reserved 6: WCN_PCM_TX (O) 7: DBG_MON_B[8] (IO)
8:6	**GPIO32_M**	GPIO32_MODE	**GPIO 32 Mode Selection** 0: GPIO32 (IO) 1: Reserved 2: Reserved 3: Reserved

Bit(s)	Mnemonic	Name	Description
			4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO31_M**	GPIO31_MODE	**GPIO 31 Mode Selection** 0: GPIO31 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO30_M**	GPIO30_MODE	**GPIO 30 Mode Selection** 0: GPIO30 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

100057D0 **GPIO_MODE8** **GPIO Mode Control Register 8** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO39_M			GPIO38_M			GPIO37_M			GPIO36_M			GPIO35_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO39_M**	GPIO39_MODE	**GPIO 39 Mode Selection** 0: GPIO39 (IO) 1: JTMS (IO) 2: CONN_MCU_TMS (I) 3: CONN_MCU_AICE_JMSC (IO) 4: DFD_TMS_XI (I) 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO38_M**	GPIO38_MODE	**GPIO 38 Mode Selection** 0: GPIO38 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO37_M**	GPIO37_MODE	**GPIO 37 Mode Selection** 0: GPIO37 (IO) 1: I2S1_MCLK (IO)

Bit(s)	Mnemonic	Name	Description
			2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[11] (IO)
5:3	GPIO36_M	GPIO36_MODE	**GPIO 36 Mode Selection** 0: GPIO36 (IO) 1: I2S1_LRCK (IO) 2: Reserved 3: PCM_SYNC (IO) 4: Reserved 5: Reserved 6: WCN_PCM_SYNC (IO) 7: DBG_MON_B[10] (IO)
2:0	GPIO35_M	GPIO35_MODE	**GPIO 35 Mode Selection** 0: GPIO35 (IO) 1: I2S1_BCK (IO) 2: Reserved 3: PCM_CLK0 (IO) 4: Reserved 5: Reserved 6: WCN_PCM_CLKO (IO) 7: DBG_MON_B[9] (IO)

100057E0 GPIO_MODE9 **GPIO Mode Control Register 9** **0049**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO44_M			GPIO43_M			GPIO42_M			GPIO41_M			GPIO40_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	GPIO44_M	GPIO44_MODE	**GPIO 44 Mode Selection** 0: GPIO44 (IO) 1: NCEB1 (O) 2: IDDIG (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	GPIO43_M	GPIO43_MODE	**GPIO 43 Mode Selection** 0: GPIO43 (IO) 1: NCLE (O) 2: SFLASH_CS_L2 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	GPIO42_	GPIO42_MODE	**GPIO 42 Mode Selection**

Bit(s)	Mnemonic	Name	Description
	M		0: GPIO42 (IO) 1: JTDO (O) 2: CONN_MCU_TDO (O) 3: Reserved 4: DFD_TDO (O) 5: Reserved 6: Reserved 7: Reserved
5:3	GPIO41_M	GPIO41_MODE	**GPIO 41 Mode Selection** 0: GPIO41 (IO) 1: JTDI (I) 2: CONN_MCU_TDI (I) 3: Reserved 4: DFD_TDI_XI (I) 5: Reserved 6: Reserved 7: Reserved
2:0	GPIO40_M	GPIO40_MODE	**GPIO 40 Mode Selection** 0: GPIO40 (IO) 1: JTCK (I) 2: CONN_MCU_TCK1 (I) 3: CONN_MCU_AICE_JCKC (I) 4: DFD_TCK_XI (I) 5: Reserved 6: Reserved 7: Reserved

100057F0 **GPIO_MODE10** **GPIO Mode Control Register 10** **1000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO49_M			GPIO48_M			GPIO47_M			GPIO46_M			GPIO45_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	GPIO49_M	GPIO49_MODE	**GPIO 49 Mode Selection** 0: GPIO49 (IO) 1: I2S0_DATA (IO) 2: I2S0_DATA_BYPS (O) 3: PCM_TX (O) 4: Reserved 5: Reserved 6: WCN_I2S_DO (O) 7: DBG_MON_B[3] (IO)
11:9	GPIO48_M	GPIO48_MODE	**GPIO 48 Mode Selection** 0: GPIO48 (IO) 1: NRNB (I) 2: DRV_VBUS_P1 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved

Bit(s)	Mnemonic	Name	Description
			7: Reserved
8:6	**GPIO47_M**	GPIO47_MODE	**GPIO 47 Mode Selection** 0: GPIO47 (IO) 1: NREB (O) 2: IDDIG_P1 (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO46_M**	GPIO46_MODE	**GPIO 46 Mode Selection** 0: GPIO46 (IO) 1: IR (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO45_M**	GPIO45_MODE	**GPIO 45 Mode Selection** 0: GPIO45 (IO) 1: NCEB0 (O) 2: DRV_VBUS (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005800 **GPIO_MODE11** **GPIO Mode Control Register 11** **0049**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO54_M			GPIO53_M			GPIO52_M			GPIO51_M			GPIO50_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO54_M**	GPIO54_MODE	**GPIO 54 Mode Selection** 0: GPIO54 (IO) 1: SPI0_CK (O) 2: Reserved 3: SPDIF_IN1 (I) 4: ADC_DAT_IN (I) 5: Reserved 6: Reserved 7: DBG_MON_A[10] (IO)
11:9	**GPIO53_M**	GPIO53_MODE	**GPIO 53 Mode Selection** 0: GPIO53 (IO) 1: SPI0_CS (O) 2: Reserved 3: SPDIF (O) 4: ADC_CK (O)

Bit(s)	Mnemonic	Name	Description
			5: PWM1 (O) 6: Reserved 7: DBG_MON_A[7] (IO)
8:6	**GPIO52_M**	GPIO52_MODE	**GPIO 52 Mode Selection** 0: GPIO52 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO51_M**	GPIO51_MODE	**GPIO 51 Mode Selection** 0: GPIO51 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO50_M**	GPIO50_MODE	**GPIO 50 Mode Selection** 0: GPIO50 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005810 **GPIO_MODE12** **GPIO Mode Control Register 12** **0240**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO59_M			GPIO58_M			GPIO57_M			GPIO56_M			GPIO55_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	1	0	0	1	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**Reserved**	Reserved	**Reserved**
11:9	**GPIO58_M**	GPIO58_MODE	**GPIO 58 Mode Selection** 0: GPIO58 (IO) 1: SCL1 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO57_M**	GPIO57_MODE	**GPIO 57 Mode Selection** 0: GPIO57 (IO)

Bit(s)	Mnemonic	Name	Description
			1: SDA1 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	GPIO56_M	GPIO56_MODE	**GPIO 56 Mode Selection** 0: GPIO56 (IO) 1: SPI0_MO (O) 2: SPI0_MI (I) 3: SPDIF_IN0 (I) 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[9] (IO)
2:0	GPIO55_M	GPIO55_MODE	**GPIO 55 Mode Selection** 0: GPIO55 (IO) 1: SPI0_MI (I) 2: SPI0_MO (O) 3: MSDC1_WP (I) 4: ADC_WS (O) 5: PWM2 (O) 6: Reserved 7: DBG_MON_A[8] (IO)

10005820 **GPIO_MODE13** **GPIO Mode Control Register 13** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO64_M			GPIO63_M			GPIO62_M			GPIO61_M			GPIO60_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	GPIO64_M	GPIO64_MODE	**GPIO 64 Mode Selection** 0: GPIO64 (IO) 1: WB_SDATA(IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[12] (IO)
11:9	GPIO63_M	GPIO63_MODE	**GPIO 63 Mode Selection** 0: GPIO63 (IO) 1: WB_SCLK(O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[13] (IO)

Bit(s)	Mnemonic	Name	Description
8:6	**GPIO62_M**	GPIO62_MODE	**GPIO 62 Mode Selection** 0: GPIO62 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[15] (IO)
5:3	**GPIO61_M**	GPIO61_MODE	**GPIO 61 Mode Selection** 0: GPIO61 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[16] (IO)
2:0	**GPIO60_M**	GPIO60_MODE	**GPIO 60 Mode Selection** 0: GPIO60 (IO) 1: WB_RSTB(O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[11] (IO)

10005830 **GPIO_MODE14** **GPIO Mode Control Register 14** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO69_M			GPIO68_M			GPIO67_M			GPIO66_M			GPIO65_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO69_M**	GPIO69_MODE	**GPIO 69 Mode Selection** 0: GPIO69 (IO) 1: WB_CTRL3(IO) 2: Reserved 3: Reserved 4: Reserved 5: DFD_TDI_XI (I) 6: Reserved 7: DBG_MON_A[20] (IO)
11:9	**GPIO68_M**	GPIO68_MODE	**GPIO 68 Mode Selection** 0: GPIO68 (IO) 1: WB_CTRL2(IO) 2: Reserved 3: Reserved 4: Reserved 5: DFD_TCK_XI (I)

Bit(s)	Mnemonic	Name	Description
			6: Reserved 7: DBG_MON_A[19] (IO)
8:6	**GPIO67_M**	GPIO67_MODE	**GPIO 67 Mode Selection** 0: GPIO67 (IO) 1: WB_CTRL1(IO) 2: Reserved 3: Reserved 4: Reserved 5: DFD_TMS_XI (I) 6: Reserved 7: DBG_MON_A[18] (IO)
5:3	**GPIO66_M**	GPIO66_MODE	**GPIO 66 Mode Selection** 0: GPIO66 (IO) 1: WBC_TRLo(IO) 2: Reserved 3: Reserved 4: Reserved 5: DFD_NTRST_XI (I) 6: Reserved 7: DBG_MON_A[17] (IO)
2:0	**GPIO65_M**	GPIO65_MODE	**GPIO 65 Mode Selection** 0: GPIO65 (IO) 1: WB_SEN(O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[14] (IO)

10005840 **GPIO_MODE15** **GPIO Mode Control Register 15** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO74_M			GPIO73_M			GPIO72_M			GPIO71_M			GPIO70_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO74_M**	GPIO74_MODE	**GPIO 74 Mode Selection** 0: GPIO74 (IO) 1: I2So_BCK (IO) 2: Reserved 3: PCM_CLKo (IO) 4: Reserved 5: Reserved 6: WCN_I2S_BCK (IO) 7: DBG_MON_B[4] (IO)
11:9	**GPIO73_M**	GPIO73_MODE	**GPIO 73 Mode Selection** 0: GPIO73 (IO) 1: I2So_LRCK (IO) 2: Reserved 3: PCM_SYNC (IO)

Bit(s)	Mnemonic	Name	Description
			4: Reserved 5: Reserved 6: WCN_I2S_LRCK (IO) 7: DBG_MON_B[5] (IO)
8:6	**GPIO72_M**	GPIO72_MODE	**GPIO 72 Mode Selection** 0: GPIO72 (IO) 1: I2S0_DATA_IN (IO) 2: Reserved 3: PCM_RX (I) 4: PWM0 (O) 5: DISP_PWM (O) 6: WCN_I2S_DI (I) 7: DBG_MON_B[2] (IO)
5:3	**GPIO71_M**	GPIO71_MODE	**GPIO 71 Mode Selection** 0: GPIO71 (IO) 1: WB_CTRL5(IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_A[22] (IO)
2:0	**GPIO70_M**	GPIO70_MODE	**GPIO 70 Mode Selection** 0: GPIO70 (IO) 1: WB_CTRL4(IO) 2: Reserved 3: Reserved 4: Reserved 5: DFD_TDO (O) 6: Reserved 7: DBG_MON_A[21] (IO)

10005850 **GPIO_MODE16** **GPIO Mode Control Register 16** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO79_M			GPIO78_M			GPIO77_M			GPIO76_M			GPIO75_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO79_M**	GPIO79_MODE	**GPIO 79 Mode Selection** 0: GPIO79 (IO) 1: URXD0 (I) 2: UTXD0 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO78_M**	GPIO78_MODE	**GPIO 78 Mode Selection** 0: GPIO78 (IO) 1: SCL2 (IO)

Bit(s)	Mnemonic	Name	Description
			2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO77_M**	GPIO77_MODE	**GPIO 77 Mode Selection** 0: GPIO77 (IO) 1: SDA2 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO76_M**	GPIO76_MODE	**GPIO 76 Mode Selection** 0: GPIO76 (IO) 1: SCL0 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO75_M**	GPIO75_MODE	**GPIO 75 Mode Selection** 0: GPIO75 (IO) 1: SDA0 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005860 **GPIO_MODE17** **GPIO Mode Control Register 17** **0001**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO84_M			GPIO83_M			GPIO82_M			GPIO81_M			GPIO80_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO84_M**	GPIO84_MODE	**GPIO 84 Mode Selection** 0: GPIO84 (IO) 1: DSI_TE (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[0] (IO)
11:9	**GPIO83_**	GPIO83_MODE	**GPIO 83 Mode Selection**

Bit(s)	Mnemonic	Name	Description
	M		0: GPIO83 (IO) 1: LCM_RST (O) 2: VDAC_CK_XI (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[1] (IO)
8:6	**GPIO82_M**	GPIO82_MODE	**GPIO 82 Mode Selection** 0: GPIO82 (IO) 1: UTXD1 (O) 2: URXD1 (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO81_M**	GPIO81_MODE	**GPIO 81 Mode Selection** 0: GPIO81 (IO) 1: URXD1 (I) 2: UTXD1 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO80_M**	GPIO80_MODE	**GPIO 80 Mode Selection** 0: GPIO80 (IO) 1: UTXD0 (O) 2: URXD0 (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005870 GPIO_MODE18 GPIO Mode Control Register 18 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO89_M			GPIO88_M			GPIO87_M			GPIO86_M			GPIO85_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO89_M**	GPIO89_MODE	**GPIO 89 Mode Selection** 0: GPIO89 (IO) 1: Reserved 2: ANT_SEL4 (O) 3: SDA2 (IO) 4: Reserved 5: UTXD1 (O) 6: PWM2 (O)

Bit(s)	Mnemonic	Name	Description
			7: Reserved
11:9	**GPIO88_M**	GPIO88_MODE	**GPIO 88 Mode Selection** 0: GPIO88 (IO) 1: Reserved 2: ANT_SEL3 (O) 3: PWM0 (O) 4: Reserved 5: URXD0 (I) 6: PWM1 (O) 7: Reserved
8:6	**GPIO87_M**	GPIO87_MODE	**GPIO 87 Mode Selection** 0: GPIO87 (IO) 1: Reserved 2: ANT_SEL2 (O) 3: Reserved 4: Reserved 5: UTXD0 (O) 6: I2SOUT_DATA_OUT (O) 7: Reserved
5:3	**GPIO86_M**	GPIO86_MODE	**GPIO 86 Mode Selection** 0: GPIO86 (IO) 1: Reserved 2: ANT_SEL1 (O) 3: SCL1 (IO) 4: Reserved 5: Reserved 6: I2SOUT_LRCK (O) 7: Reserved
2:0	**GPIO85_M**	GPIO85_MODE	**GPIO 85 Mode Selection** 0: GPIO85 (IO) 1: Reserved 2: ANT_SEL0 (O) 3: SDA1 (IO) 4: Reserved 5: Reserved 6: I2SOUT_BCK (O) 7: Reserved

10005880 **GPIO_MODE19** **GPIO Mode Control Register 19** **1248**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO94_M			GPIO93_M			GPIO92_M			GPIO91_M			GPIO90_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**Reserved**	Reserved	**Reserved**

10005890 **GPIO_MODE 20** **GPIO Mode Control Register 20** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO99_M			GPIO98_M			GPIO97_M			GPIO96_M			GPIO95_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO99_M**	GPIO99_MODE	**GPIO 99 Mode Selection** 0: GPIO99 (IO) 1: MIPI_TDN0 (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO98_M**	GPIO98_MODE	**GPIO 98 Mode Selection** 0: GPIO98 (IO) 1: MIPI_TDP1 (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO97_M**	GPIO97_MODE	**GPIO 97 Mode Selection** 0: GPIO97 (IO) 1: MIPI_TDN1 (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO96_M**	GPIO96_MODE	**GPIO 96 Mode Selection** 0: GPIO96 (IO) 1: MIPI_TCP (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO95_M**	GPIO95_MODE	**GPIO 95 Mode Selection** 0: GPIO95 (IO) 1: MIPI_TCN (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

100058A0 **GPIO_MODE21** **GPIO Mode Control Register 21** **0001**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO104_M			GPIO103_M			GPIO102_M			GPIO101_M			GPIO100_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO104_M**	GPIO104_MODE	**GPIO 104 Mode Selection** 0: GPIO104 (IO) 1: SPI2_CK (O) 2: Reserved 3: Reserved 4: KROW3 (IO) 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO103_M**	GPIO103_MODE	**GPIO 103 Mode Selection** 0: GPIO103 (IO) 1: SPI2_MO (O) 2: SPI2_MI (I) 3: Reserved 4: KROW2 (IO) 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO102_M**	GPIO102_MODE	**GPIO 102 Mode Selection** 0: GPIO102 (IO) 1: SPI2_MI (I) 2: SPI2_MO (O) 3: Reserved 4: KROW1 (IO) 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO101_M**	GPIO101_MODE	**GPIO 101 Mode Selection** 0: GPIO101 (IO) 1: SPI2_CS (O) 2: Reserved 3: Reserved 4: KROW0 (IO) 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO100_M**	GPIO100_MODE	**GPIO 100 Mode Selection** 0: GPIO100 (IO) 1: MIPI_TDP0 (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

100058B0 **GPIO_MODE22** **GPIO Mode Control Register 22** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO109_M			GPIO108_M			GPIO107_M			GPIO106_M			GPIO105_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO109_M**	GPIO109_MODE	**GPIO 109 Mode Selection** 0: GPIO109 (IO) 1: MSDC1_DAT2 (IO) 2: ANT_SEL4 (O) 3: SDA2 (IO) 4: Reserved 5: UTXD1 (O) 6: PWM2 (O) 7: DBG_MON_B[24] (IO)
11:9	**GPIO108_M**	GPIO108_MODE	**GPIO 108 Mode Selection** 0: GPIO108 (IO) 1: MSDC1_DAT1 (IO) 2: ANT_SEL3 (O) 3: PWM0 (O) 4: Reserved 5: URXD0 (I) 6: PWM1 (O) 7: DBG_MON_B[25] (IO)
8:6	**GPIO107_M**	GPIO107_MODE	**GPIO 107 Mode Selection** 0: GPIO107 (IO) 1: MSDC1_DAT0 (IO) 2: ANT_SEL2 (O) 3: Reserved 4: Reserved 5: UTXD0 (O) 6: I2SOUT_DATA_OUT (O) 7: DBG_MON_B[26] (IO)
5:3	**GPIO106_M**	GPIO106_MODE	**GPIO 106 Mode Selection** 0: GPIO106 (IO) 1: MSDC1_CLK (O) 2: ANT_SEL1 (O) 3: SCL1 (IO) 4: Reserved 5: Reserved 6: I2SOUT_LRCK (O) 7: DBG_MON_B[28] (IO)
2:0	**GPIO105_M**	GPIO105_MODE	**GPIO 105 Mode Selection** 0: GPIO105 (IO) 1: MSDC1_CMD (IO) 2: ANT_SEL0 (O) 3: SDA1 (IO) 4: Reserved 5: Reserved 6: I2SOUT_BCK (O) 7: DBG_MON_B[27] (IO)

100058C0 **GPIO_MODE23** **GPIO Mode Control Register 23** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO114_M			GPIO113_M			GPIO112_M			GPIO111_M			GPIO110_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO114_M**	GPIO114_MODE	**GPIO 114 Mode Selection** 0: GPIO114 (IO) 1: MSDC0_DAT4 (IO) 2: Reserved 3: Reserved 4: NLD4 (IO) 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO113_M**	GPIO113_MODE	**GPIO 113 Mode Selection** 0: GPIO113 (IO) 1: MSDC0_DAT5 (IO) 2: Reserved 3: Reserved 4: NLD5 (IO) 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO112_M**	GPIO112_MODE	**GPIO 112 Mode Selection** 0: GPIO112 (IO) 1: MSDC0_DAT6 (IO) 2: Reserved 3: Reserved 4: NLD6 (IO) 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO111_M**	GPIO111_MODE	**GPIO 111 Mode Selection** 0: GPIO111 (IO) 1: MSDC0_DAT7 (IO) 2: Reserved 3: Reserved 4: NLD7 (IO) 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO110_M**	GPIO110_MODE	**GPIO 110 Mode Selection** 0: GPIO110 (IO) 1: MSDC1_DAT3 (IO) 2: ANT_SEL5 (O) 3: SCL2 (IO) 4: EXT_FRAME_SYNC (I) 5: URXD1 (I) 6: PWM3 (O) 7: DBG_MON_B[23] (IO)

100058D0 **GPIO_MODE24** **GPIO Mode Control Register 24** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO119_M			GPIO118_M			GPIO117_M			GPIO116_M			GPIO115_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO119_M**	GPIO119_MODE	**GPIO 119 Mode Selection** 0: GPIO119 (IO) 1: MSDC0_DAT2 (IO) 2: Reserved 3: Reserved 4: NLD2 (IO) 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO118_M**	GPIO118_MODE	**GPIO 118 Mode Selection** 0: GPIO118 (IO) 1: MSDC0_DAT3 (IO) 2: Reserved 3: Reserved 4: NLD3 (IO) 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO117_M**	GPIO117_MODE	**GPIO 117 Mode Selection** 0: GPIO117 (IO) 1: MSDC0_CLK (O) 2: Reserved 3: Reserved 4: NWEB (O) 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO116_M**	GPIO116_MODE	**GPIO 116 Mode Selection** 0: GPIO116 (IO) 1: MSDC0_CMD (IO) 2: Reserved 3: Reserved 4: NALE (O) 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO115_M**	GPIO115_MODE	**GPIO 115 Mode Selection** 0: GPIO115 (IO) 1: MSDC0_RSTB (O) 2: Reserved 3: Reserved 4: NLD8 (IO) 5: Reserved 6: Reserved 7: Reserved

100058E0 **GPIO_MODE25** **GPIO Mode Control Register 25** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO124_M			GPIO123_M			GPIO122_M			GPIO121_M			GPIO120_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO124_M**	GPIO124_MODE	**GPIO 124 Mode Selection** 0: GPIO124 (IO) 1: HDMISCK (IO) 2: Reserved 3: Reserved 4: SDA1 (IO) 5: PWM3 (O) 6: Reserved 7: Reserved
11:9	**GPIO123_M**	GPIO123_MODE	**GPIO 123 Mode Selection** 0: GPIO123 (IO) 1: HTPLG (I) 2: Reserved 3: Reserved 4: SCL2 (IO) 5: UTXD0 (O) 6: Reserved 7: Reserved
8:6	**GPIO122_M**	GPIO122_MODE	**GPIO 122 Mode Selection** 0: GPIO122 (IO) 1: CEC (IO) 2: Reserved 3: Reserved 4: SDA2 (IO) 5: URXD0 (I) 6: Reserved 7: Reserved
5:3	**GPIO121_M**	GPIO121_MODE	**GPIO 121 Mode Selection** 0: GPIO121 (IO) 1: MSDC0_DAT0 (IO) 2: Reserved 3: Reserved 4: NLD0 (IO) 5: WATCHDOG (O) 6: Reserved 7: Reserved
2:0	**GPIO120_M**	GPIO120_MODE	**GPIO 120 Mode Selection** 0: GPIO120 (IO) 1: MSDC0_DAT1 (IO) 2: Reserved 3: Reserved 4: NLD1 (IO) 5: Reserved 6: Reserved 7: Reserved

100058F0 **GPIO_MODE 26** **GPIO Mode Control Register 26** **0009**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO129_M			GPIO128_M			GPIO127_M			GPIO126_M			GPIO125_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**Reserved**	Reserved	**Reserved**
11:9	**Reserved**	Reserved	**Reserved**
8:6	**Reserved**	Reserved	**Reserved**
5:3	**GPIO126_M**	GPIO126_MODE	**GPIO 126 Mode Selection** 0: GPIO126 (IO) 1: I2S0_MCLK (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: WCN_I2S_MCLK (O) 7: DBG_MON_B[6] (IO)
2:0	**GPIO125_M**	GPIO125_MODE	**GPIO 125 Mode Selection** 0: GPIO125 (IO) 1: HDMISD (IO) 2: Reserved 3: Reserved 4: SCL1 (IO) 5: PWM4 (O) 6: Reserved 7: Reserved

10005900 **GPIO_MODE 27** **GPIO Mode Control Register 27** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO134_M			GPIO133_M			GPIO132_M			GPIO131_M			GPIO130_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005910 **GPIO_MODE 28** **GPIO Mode Control Register 28** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO139_M			GPIO138_M			GPIO137_M			GPIO136_M			GPIO135_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005920 **GPIO_MODE29** **GPIO Mode Control Register 29** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO144_M			GPIO143_M			GPIO142_M			GPIO141_M			GPIO140_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005930 **GPIO_MODE30** **GPIO Mode Control Register 30** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO149_M			GPIO148_M			GPIO147_M			GPIO146_M			GPIO145_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005940 **GPIO_MODE31** **GPIO Mode Control Register 31** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO154_M			GPIO153_M			GPIO152_M			GPIO151_M			GPIO150_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005950 **GPIO_MODE32** **GPIO Mode Control Register 32** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10005950 GPIO_MODE 32 — **GPIO Mode Control Register 32** — **0000**

Name		GPIO159_M			GPIO158_M			GPIO157_M			GPIO156_M			GPIO155_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005960 GPIO_MODE 33 — **GPIO Mode Control Register 33** — **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO164_M			GPIO163_M			GPIO162_M			GPIO161_M			GPIO160_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005970 GPIO_MODE 34 — **GPIO Mode Control Register 34** — **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO169_M			GPIO168_M			GPIO167_M			GPIO166_M			GPIO165_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005980 GPIO_MODE 35 — **GPIO Mode Control Register 35** — **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO174_M			GPIO173_M			GPIO172_M			GPIO171_M			GPIO170_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005990 **GPIO_MODE36** **GPIO Mode Control Register 36** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO179_M			GPIO178_M			GPIO177_M			GPIO176_M			GPIO175_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

100059A0 **GPIO_MODE37** **GPIO Mode Control Register 37** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO184_M			GPIO183_M			GPIO182_M			GPIO181_M			GPIO180_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

100059B0 **GPIO_MODE38** **GPIO Mode Control Register 38** **1200**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO189_M			GPIO188_M			GPIO187_M			GPIO186_M			GPIO185_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO189_M**	GPIO189_MODE	**GPIO 189 Mode Selection** 0: GPIO189 (IO) 1: Reserved 2:Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO188_M**	GPIO188_MODE	**GPIO 188 Mode Selection** 0: GPIO188 (IO) 1: Reserved 2: Reserved 3: Reserved

Bit(s)	Mnemonic	Name	Description
			4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**Reserved**	Reserved	**Reserved**
5:3	**Reserved**	Reserved	**Reserved**
2:0	**Reserved**	Reserved	**Reserved**

100059C0 GPIO_MODE39 GPIO Mode Control Register 39 1249

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO194_M			GPIO193_M			GPIO192_M			GPIO191_M			GPIO190_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO194_M**	GPIO194_MODE	**GPIO 194 Mode Selection** 0: GPIO194 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO193_M**	GPIO193_MODE	**GPIO 193 Mode Selection** 0: GPIO193 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO192_M**	GPIO192_MODE	**GPIO 192 Mode Selection** 0: GPIO192 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO191_M**	GPIO191_MODE	**GPIO 191 Mode Selection** 0: GPIO191 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved

Bit(s)	Mnemonic	Name	Description
			7: Reserved
2:0	**GPIO190_M**	GPIO190_MODE	**GPIO 190 Mode Selection** 0: GPIO190 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

100059D0 **GPIO_MODE40** **GPIO Mode Control Register 40** **0249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO199_M			GPIO198_M			GPIO197_M			GPIO196_M			GPIO195_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO199_M**	GPIO199_MODE	**GPIO 199 Mode Selection** 0: GPIO199 (IO) 1: SPI1_CK (O) 2: Reserved 3: EXT_FRAME_SYNC (I) 4: KCOL3 (IO) 5: Reserved 6: Reserved 7: DBG_MON_B[15] (IO)
11:9	**GPIO198_M**	GPIO198_MODE	**GPIO 198 Mode Selection** 0: GPIO198 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO197_M**	GPIO197_MODE	**GPIO 197 Mode Selection** 0: GPIO197 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO196_M**	GPIO196_MODE	**GPIO 196 Mode Selection** 0: GPIO196 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved

Bit(s)	Mnemonic	Name	Description
			5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO195_M**	GPIO195_MODE	**GPIO 195 Mode Selection** 0: GPIO195 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

100059E0 **GPIO_MODE41** **GPIO Mode Control Register 41** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO204_M			GPIO203_M			GPIO202_M			GPIO201_M			GPIO200_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO204_M**	GPIO204_MODE	**GPIO 204 Mode Selection** 0: GPIO204 (IO) 1: PWM1 (O) 2: CLKM3 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[19] (IO)
11:9	**GPIO203_M**	GPIO203_MODE	**GPIO 203 Mode Selection** 0: GPIO203 (IO) 1: PWM0 (O) 2: DISP_PWM (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[18] (IO)
8:6	**GPIO202_M**	GPIO202_MODE	**GPIO 202 Mode Selection** 0: GPIO202 (IO) 1: SPDIF_IN1 (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO201_M**	GPIO201_MODE	**GPIO 201 Mode Selection** 0: GPIO201 (IO) 1: SPDIF_IN0 (I) 2: Reserved

Bit(s)	Mnemonic	Name	Description
			3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[17] (IO)
2:0	**GPIO200_M**	GPIO200_MODE	**GPIO 200 Mode Selection** 0: GPIO200 (IO) 1: SPDIF_OUT (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[16] (IO)

100059F0 **GPIO_MODE42** **GPIO Mode Control Register 42** **1200**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO209_M			GPIO208_M			GPIO207_M			GPIO206_M			GPIO205_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO209_M**	GPIO209_MODE	**GPIO 209 Mode Selection** 0: GPIO209 (IO) 1: AUD_EXT_CK2 (I) 2: MSDC1_WP (I) 3: Reserved 4: Reserved 5: PWM1 (O) 6: Reserved 7: DBG_MON_A[32] (IO)
11:9	**GPIO208_M**	GPIO208_MODE	**GPIO 208 Mode Selection** 0: GPIO208 (IO) 1: AUD_EXT_CK1 (I) 2: PWM0 (O) 3: Reserved 4: ANT_SEL5 (O) 5: DISP_PWM (O) 6: Reserved 7: DBG_MON_A[31] (IO)
8:6	**GPIO207_M**	GPIO207_MODE	**GPIO 207 Mode Selection** 0: GPIO207 (IO) 1: PWM4 (O) 2: CLKM0 (O) 3: EXT_FRAME_SYNC (I) 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[22] (IO)
5:3	**GPIO206_M**	GPIO206_MODE	**GPIO 206 Mode Selection** 0: GPIO206 (IO)

Bit(s)	Mnemonic	Name	Description
			1: PWM3 (O) 2: CLKM1 (O) 3: EXT_FRAME_SYNC (I) 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[21] (IO)
2:0	**GPIO205_M**	GPIO205_MODE	**GPIO 205 Mode Selection** 0: GPIO205 (IO) 1: PWM2 (O) 2: CLKM2 (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[20] (IO)

10005A00 **GPIO_MODE43** **GPIO Mode Control Register 43** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO214_M			GPIO213_M			GPIO212_M			GPIO211_M			GPIO210_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005A10 **GPIO_MODE44** **GPIO Mode Control Register 44** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO219_M			GPIO218_M			GPIO217_M			GPIO216_M			GPIO215_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005A20 **GPIO_MODE45** **GPIO Mode Control Register 45** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO224_M			GPIO223_M			GPIO222_M			GPIO221_M			GPIO220_M		
Type		RW			RW			RW			RW			RW		

10005A20 **GPIO_MODE45** **GPIO Mode Control Register 45** **0000**

Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005A30 **GPIO_MODE46** **GPIO Mode Control Register 46** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO229_M			GPIO228_M			GPIO227_M			GPIO226_M			GPIO225_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005A40 **GPIO_MODE47** **GPIO Mode Control Register 47** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO234_M			GPIO233_M			GPIO232_M			GPIO231_M			GPIO230_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:0	**Reserved**	Reserved	**Reserved**

10005A50 **GPIO_MODE48** **GPIO Mode Control Register 48** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO239_M			GPIO238_M			GPIO237_M			GPIO236_M			GPIO235_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO239_M**	GPIO239_MODE	**GPIO 239 Mode Selection** 0: GPIO239 (IO) 1: SFLASH_IO_0 (IO) 2: DRV_VBUS_P1 (O)

Bit(s)	Mnemonic	Name	Description
			3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO238_M**	GPIO238_MODE	**GPIO 238 Mode Selection** 0: GPIO238 (IO) 1: SFLASH_IO_1 (IO) 2: IDDIG_P1 (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO237_M**	GPIO237_MODE	**GPIO 237 Mode Selection** 0: GPIO237 (IO) 1: SFLASH_IO_2 (IO) 2: DRV_VBUS (O) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO236_M**	GPIO236_MODE	**GPIO 236 Mode Selection** 0: GPIO236 (IO) 1: SFLASH_IO_3 (IO) 2: IDDIG (I) 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**Reserved**	Reserved	**Reserved**

10005A60 **GPIO_MODE49** **GPIO Mode Control Register 49** **1000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO244_M			GPIO243_M			GPIO242_M			GPIO241_M			GPIO240_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO244_M**	GPIO244_MODE	**GPIO 244 Mode Selection** 0: GPIO244 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

Bit(s)	Mnemonic	Name	Description
11:9	**GPIO243_M**	GPIO243_MODE	**GPIO 243 Mode Selection** 0: GPIO243 (IO) 1: UCTS2 (I) 2: URXD3 (I) 3: UTXD3 (O) 4: SDA1 (IO) 5: Reserved 6: Reserved 7: DBG_MON_A[6] (IO)
8:6	**GPIO242_M**	GPIO242_MODE	**GPIO 242 Mode Selection** 0: GPIO242 (IO) 1: URTS2 (O) 2: UTXD3 (O) 3: URXD3 (I) 4: SCL1 (IO) 5: Reserved 6: Reserved 7: DBG_MON_B[32] (IO)
5:3	**GPIO241_M**	GPIO241_MODE	**GPIO 241 Mode Selection** 0: GPIO241 (IO) 1: SFLASH_CLK (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO240_M**	GPIO240_MODE	**GPIO 240 Mode Selection** 0: GPIO240 (IO) 1: SFLASH_CS_L (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005A70 GPIO_MODE50 GPIO Mode Control Register 50 1241

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO249_M			GPIO248_M			GPIO247_M			GPIO246_M			GPIO245_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO249_M**	GPIO249_MODE	**GPIO 249 Mode Selection** 0: GPIO249 (IO) 1:Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved

Bit(s)	Mnemonic	Name	Description
			6: Reserved 7: Reserved
11:9	**GPIO248_M**	GPIO248_MODE	**GPIO 248 Mode Selection** 0: GPIO248 (IO) 1: HDMI_TESTOUTP_RX (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO247_M**	GPIO247_MODE	**GPIO 247 Mode Selection** 0: GPIO247 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO246_M**	GPIO246_MODE	**GPIO 246 Mode Selection** 0: GPIO246 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO245_M**	GPIO245_MODE	**GPIO 245 Mode Selection** 0: GPIO245 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005A80 **GPIO_MODE51** **GPIO Mode Control Register 51** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO254_M			GPIO253_M			GPIO252_M			GPIO251_M			GPIO250_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO254_M**	GPIO254_MODE	**GPIO 254 Mode Selection** 0: GPIO254 (IO) 1: Reserved 2: Reserved 3: Reserved

Bit(s)	Mnemonic	Name	Description
			4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO253_M**	GPIO253_MODE	**GPIO 253 Mode Selection** 0: GPIO253 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO252_M**	GPIO252_MODE	**GPIO 252 Mode Selection** 0: GPIO252 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO251_M**	GPIO251_MODE	**GPIO 251 Mode Selection** 0: GPIO251 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO250_M**	GPIO250_MODE	**GPIO 250 Mode Selection** 0: GPIO250 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005A90 **GPIO_MODE52** **GPIO Mode Control Register 52** **1249**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO259_M			GPIO258_M			GPIO257_M			GPIO256_M			GPIO255_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	1	0	0	1	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO259_M**	GPIO259_MODE	**GPIO 259 Mode Selection** 0: GPIO259 (IO) 1: Reserved

Bit(s)	Mnemonic	Name	Description
			2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO258_M**	GPIO258_MODE	**GPIO 258 Mode Selection** 0: GPIO258 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO257_M**	GPIO257_MODE	**GPIO 257 Mode Selection** 0: GPIO257 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO256_M**	GPIO256_MODE	**GPIO 256 Mode Selection** 0: GPIO256 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO255_M**	GPIO255_MODE	**GPIO 255 Mode Selection** 0: GPIO255 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005AA0 **GPIO_MODE53** **GPIO Mode Control Register 53** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO264_M			GPIO263_M			GPIO262_M			GPIO261_M			GPIO260_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO264**	GPIO264_MODE	**GPIO 264 Mode Selection**

Bit(s)	Mnemonic	Name	Description
	_M		0: GPIO264 (IO) 1: GE2_TXD2 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL4 (O) 7: Reserved
11:9	**GPIO263_M**	GPIO263_MODE	**GPIO 263 Mode Selection** 0: GPIO263 (IO) 1: GE2_TXD3 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL5 (O) 7: Reserved
8:6	**GPIO262_M**	GPIO262_MODE	**GPIO 262 Mode Selection** 0: GPIO262 (IO) 1: GE2_TXEN (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO261_M**	GPIO261_MODE	**GPIO 261 Mode Selection** 0: GPIO261 (IO) 1: MSDC1_INS (I) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: DBG_MON_B[29] (IO)
2:0	**GPIO260_M**	GPIO260_MODE	**GPIO 260 Mode Selection** 0: GPIO260 (IO) 1: Reserved 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005AB0 **GPIO_MODE54** **GPIO Mode Control Register 54** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO269_M			GPIO268_M			GPIO267_M			GPIO266_M			GPIO265_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO269_M**	GPIO269_MODE	**GPIO 269 Mode Selection** 0: GPIO269 (IO) 1: GE2_RXD0 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**GPIO268_M**	GPIO268_MODE	**GPIO 268 Mode Selection** 0: GPIO268 (IO) 1: GE2_RXCLK (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**GPIO267_M**	GPIO267_MODE	**GPIO 267 Mode Selection** 0: GPIO267 (IO) 1: GE2_TXCLK (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO266_M**	GPIO266_MODE	**GPIO 266 Mode Selection** 0: GPIO266 (IO) 1: GE2_TXD0 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL2 (O) 7: Reserved
2:0	**GPIO265_M**	GPIO265_MODE	**GPIO 265 Mode Selection** 0: GPIO265 (IO) 1: GE2_TXD1 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL3 (O) 7: Reserved

10005AC0 **GPIO_MODE55** **GPIO Mode Control Register 55** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO274_M			GPIO273_M			GPIO272_M			GPIO271_M			GPIO270_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**GPIO274_M**	GPIO274_MODE	**GPIO 274 Mode Selection** 0: GPIO274 (IO) 1: GE2_RXDV (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
11:9	**Reserved**	Reserved	**Reserved**
8:6	**GPIO272_M**	GPIO272_MODE	**GPIO 272 Mode Selection** 0: GPIO272 (IO) 1: GE2_RXD3 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
5:3	**GPIO271_M**	GPIO271_MODE	**GPIO 271 Mode Selection** 0: GPIO271 (IO) 1: GE2_RXD2 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
2:0	**GPIO270_M**	GPIO270_MODE	**GPIO 270 Mode Selection** 0: GPIO270 (IO) 1: GE2_RXD1 (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved

10005AD0 **GPIO_MODE56** **GPIO Mode Control Register 56** **1000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		GPIO279_M			GPIO278_M			GPIO277_M			GPIO276_M			GPIO275_M		
Type		RW			RW			RW			RW			RW		
Reset		0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	-	-	-
11:9	**GPIO278_M**	GPIO278_MODE	**GPIO 278 Mode Selection** 0: GPIO278 (IO) 1: JTAG_RESET (I) 2: Reserved

Bit(s)	Mnemonic	Name	Description
			3: Reserved 4: Reserved 5: Reserved 6: Reserved 7: Reserved
8:6	**Reserved**	Reserved	**Reserved**
5:3	**GPIO276_M**	GPIO276_MODE	**GPIO 276 Mode Selection** 0: GPIO276 (IO) 1: MDIO (IO) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL1 (O) 7: Reserved
2:0	**GPIO275_M**	GPIO275_MODE	**GPIO 275 Mode Selection** 0: GPIO275 (IO) 1: MDC (O) 2: Reserved 3: Reserved 4: Reserved 5: Reserved 6: ANT_SEL0 (O) 7: Reserved

10005B10 GPIO_BANK GPIO Misc Control Register F801

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GPIO_MISC_REG													CPUM_MIPI_GPI_EN	GPS_SYNC_SEL	GPIO_BANK
Type	RW				RO	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:3		GPIO_MISC_REG	**GPIO Reserved Control register** 0: Reserved 1: Reserved
2		CPUM_MIPI_GPI_EN	0: Disable 1: Enable
1		GPS_SYNC_SEL	0: internal frame sync 1: external frame sync
0		GPIO_BANK	**GPIO Reserved Control register** 0: Reserved 1: Reserved

10005B20 **IES_EN0** **GPIO IES Control Register 0** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IES0															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0		IES0	**GPIO Input Enable Control register** 0: Disable 1: Enable

10005B30 **IES_EN1** **GPIO IES Control Register 1** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IES1															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0		IES1	**GPIO Input Enable Control register** 0: Disable 1: Enable

10005B40 **IES_EN2** **GPIO IES Control Register 2** **FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IES2															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
15:0		IES2	**GPIO Input Enable Control register** 0: Disable 1: Enable

10005B50 **SMT_EN0** **GPIO SMT Control Register 0** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SMT0															
Type	RW															

10005B50 SMT_EN0 GPIO SMT Control Register 0 0000

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		SMT0	**GPIO Schmitter Trigger Control Register** 0: Disable 1: Enable

10005B60 SMT_EN1 GPIO SMT Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SMT1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		SMT1	**GPIO Schmitter Trigger Control Register** 0: Disable 1: Enable

10005B70 SMT_EN2 GPIO SMT Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SMT2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		SMT2	**GPIO Schmitter Trigger Control Register** 0: Disable 1: Enable

10005B80 TDSEL0 GPIO TDSEL Control Register 0 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL0	TDSEL control register

10005B90 TDSEL1 GPIO TDSEL Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL1	TDSEL control register

10005BA0 TDSEL2 GPIO TDSEL Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL2	TDSEL control register

10005BB0 TDSEL3 GPIO TDSEL Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL3															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL3	TDSEL control register

10005BC0 TDSEL4 GPIO TDSEL Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL4															

10005BC0 **TDSEL4** **GPIO TDSEL Control Register 4** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL4	TDSEL control register

10005BD0 **TDSEL5** **GPIO TDSEL Control Register 5** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TDSEL5															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		TDSEL5	TDSEL control register

10005BE0 **OD33_CTRL4** **18OD33 IO Group TDSEL/RDSEL Control Register 4** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP4						OD33_RDSEL4						OD33_TDSEL4			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP4	OD33 Control Reserve Registers
9:4		OD33_RDSEL4	RDSEL control register for eint12_tdsel
3:0		OD33_TDSEL4	TDSEL control register for eint12_tdsel

10005BF0 **OD33_CTRL5** **18OD33 IO Group TDSEL/RDSEL Control Register 5** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP5						OD33_RDSEL5						OD33_TDSEL5			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP5	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL5	**RDSEL control register for eint17_tdsel**
3:0		OD33_TDSEL5	**TDSEL control register for eint17_tdsel**

10005C00 OD33_CTRL6 18OD33 IO Group TDSEL/RDSEL Control Register 6 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP6						OD33_RDSEL6						OD33_TDSEL6			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP6	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL6	**RDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**
3:0		OD33_TDSEL6	**TDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**

10005C10 OD33_CTRL7 18OD33 IO Group TDSEL/RDSEL Control Register 7 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP7						OD33_RDSEL7						OD33_TDSEL7			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP7	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL7	**RDSEL control register for eint12_tdsel**
3:0		OD33_TDSEL7	**TDSEL control register for eint12_tdsel**

10005C20 RDSEL0 GPIO RDSEL Control Register 0 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL0	**RDSEL control register**

10005C30 RDSEL1 GPIO RDSEL Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL1	RDSEL control register

10005C40 RDSEL2 GPIO RDSEL Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL2	RDSEL control register

10005C50 RDSEL3 GPIO RDSEL Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL3															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL3	RDSEL control register

10005C60 RDSEL4 GPIO RDSEL Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL4															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL4	RDSEL control register

10005C70 RDSEL5 GPIO RDSEL Control Register5 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDSEL5															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		RDSEL5	RDSEL control register

10005C80 DRVNo_EN GPIO Control DDR Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRVNo_EN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		Reserved	Reserved

10005C90 MSDC3_CTRL0 MSDC 3 CLK Pad Control Register 0314

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3CK_BACKUP1				MS3CK_SMT	MS3CK_Ro	MS3CK_R1	MS3CK_PUPD	MS3CK_BACKUPo			MS3CK_IES	MS3CK_SR	MS3CK_E8	MS3CK_E4	MS3CK_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15:12		MS3CK_BACKUP1	Reserved
11		MS3CK_SMT	Schmitter Trigger Control 0:Disable 1:Enable
10		MS3CK_Ro	10K resistot control
9		MS3CK_R1	50K resistor control
8		MS3CK_PUPD	pull-up(0)/pull-down(1) control
7:5		MS3CK_BACKUPo	Reserved
4		MS3CK_IES	Input enable control

Bit(s)	Mnemonic	Name	Description
3		MS3CK_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS3CK_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005CA0 **DRVP0_EN** **GPIO Control DDR Register** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRVP0_EN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		Reserved	**Reserved**

10005CB0 **MSDC3_CTRL1** **MSDC 3 CMD Pad Control Register** **0314**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3CMD_BACKUP1				MS3CMD_SMT	MS3CMD_R0	MS3CMD_R1	MS3CMD_PUPD	MS3CMD_BACKUP0			MS3CMD_IES	MS3CMD_SR	MS3CMD_E8	MS3CMD_E4	MS3CMD_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15:12		MS3CMD_BACKUP1	**Reserved**
11		MS3CMD_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS3CMD_R0	**10K resistot control**
9		MS3CMD_R1	**50K resistor control**
8		MS3CMD_PUPD	**pull-up(0)/pull-down(1) control**

Bit(s)	Mnemonic	Name	Description
7:5		MS3CMD_BACKUP0	**Reserved**
4		MS3CMD_IES	**Input enable control**
3		MS3CMD_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS3CMD_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005CC0 **MSDC0_CTRL0** **MSDC 0 CLK Pad Control Register** **0311**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSOCK_BACKUP1				MSOCK_SMT	MSOCK_R0	MSOCK_R1	MSOCK_PUPD	MSOCK_BACKUP0			MSOCK_IES	MSOCK_SR	MSOCK_E8	MSOCK_E4	MSOCK_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MSOCK_BACKUP1	**Reserved**
11		MSOCK_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MSOCK_R0	**10K resistot control**
9		MSOCK_R1	**50K resistor control**
8		MSOCK_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MSOCK_BACKUP0	**Reserved**
4		MSOCK_IES	**Input enable control**
3		MSOCK_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MSOCK_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA

Bit(s)	Mnemonic	Name	Description
			100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005CD0 **MSDC0_CTRL1** **MSDC 0 CMD Pad Control Register** **0411**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSoCMD_BACKUP1				MSoCMD_SMT	MSoCMD_R0	MSoCMD_R1	MSoCMD_PUPD	MSoCMD_BACKUP0			MSoCMD_IES	MSoCMD_SR	MSoCMD_E8	MSoCMD_E4	MSoCMD_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MSoCMD_BACKUP1	**Reserved**
11		MSoCMD_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MSoCMD_R0	**10K resistot control**
9		MSoCMD_R1	**50K resistor control**
8		MSoCMD_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MSoCMD_BACKUP0	**Reserved**
4		MSoCMD_IES	**Input enable control**
3		MSoCMD_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MSoCMD_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005CE0 **MSDC0_CTRL2** **MSDC 0 DATA Pad Control Register 0** **0011**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	MSoDAT_BACKUP											MS	MS	MS	MS	MS

10005CE0 **MSDC0_CTRL2** **MSDC 0 DATA Pad Control Register 0** **0011**

e												oDAT_IES	oDAT_SR	oDAT_E8	oDAT_E4	oDAT_E2
Type	RW											RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:5		MSODAT_BACKUP	**Reserved**
4		MSODAT_IES	**Input enable control**
3		MSODAT_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MSODAT_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005CF0 **MSDC0_CTRL3** **MSDC 0 DATA Pad Control Register 1** **4444**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSODAT3_SMT	MSODAT3_R0	MSODAT3_R1	MSODAT3_PUPD	MSODAT2_SMT	MSODAT2_R0	MSODAT2_R1	MSODAT2_PUPD	MSODAT1_SMT	MSODAT1_R0	MSODAT1_R1	MSODAT1_PUPD	MSODAT0_SMT	MSODAT0_R0	MSODAT0_R1	MSODAT0_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15		MSODAT3_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
14		MSODAT3_R0	**10K resistot control**
13		MSODAT3_R1	**50K resistor control**
12		MSODAT3_PUPD	**pull-up(0)/pull-down(1) control**
11		MSODAT2_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable

Bit(s)	Mnemonic	Name	Description
10		MSoDAT2_Ro	10K resistot control
9		MSoDAT2_R1	50K resistor control
8		MSoDAT2_PUPD	pull-up(0)/pull-down(1) control
7		MSoDAT1_SMT	Schmitter Trigger Control 0:Disable 1:Enable
6		MSoDAT1_Ro	10K resistot control
5		MSoDAT1_R1	50K resistor control
4		MSoDAT1_PUPD	pull-up(0)/pull-down(1) control
3		MSoDAT0_SMT	Schmitter Trigger Control 0:Disable 1:Enable
2		MSoDAT0_Ro	10K resistot control
1		MSoDAT0_R1	50K resistor control
0		MSoDAT0_PUPD	pull-up(0)/pull-down(1) control

10005D00 **MSDC0_CTRL4** **MSDC 0 DATA Pad Control Register 2** **4444**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSoDAT7_SMT	MSoDAT7_Ro	MSoDAT7_R1	MSoDAT7_PUPD	MSoDAT6_SMT	MSoDAT6_Ro	MSoDAT6_R1	MSoDAT6_PUPD	MSoDAT5_SMT	MSoDAT5_Ro	MSoDAT5_R1	MSoDAT5_PUPD	MSoDAT4_SMT	MSoDAT4_Ro	MSoDAT4_R1	MSoDAT4_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15		MSoDAT7_SMT	Schmitter Trigger Control 0:Disable 1:Enable
14		MSoDAT7_Ro	10K resistot control
13		MSoDAT7_R1	50K resistor control
12		MSoDAT7_PUPD	pull-up(0)/pull-down(1) control
11		MSoDAT6_SMT	Schmitter Trigger Control 0:Disable 1:Enable
10		MSoDAT6_Ro	10K resistot control
9		MSoDAT6_R1	50K resistor control
8		MSoDAT6_PUPD	pull-up(0)/pull-down(1) control
7		MSoDAT5_SMT	Schmitter Trigger Control 0:Disable

Bit(s)	Mnemonic	Name	Description
			1:Enable
6		MSoDAT5_Ro	**10K resistot control**
5		MSoDAT5_R1	**50K resistor control**
4		MSoDAT5_PUPD	**pull-up(0)/pull-down(1) control**
			Schmitter Trigger Control
3		MSoDAT4_SMT	0:Disable 1:Enable
2		MSoDAT4_Ro	**10K resistot control**
1		MSoDAT4_R1	**50K resistor control**
0		MSoDAT4_PUPD	**pull-up(0)/pull-down(1) control**

10005D10 MSDCo_CTRL5 MSDC o DATA Pad Control Register 3 0004

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSoDAT_BACKUP												MSoRSTB_SMT	MSoRSTB_Ro	MSoRSTB_R1	MSoRSTB_PUPD
Type	RW												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15:4		MSoDAT_BACKUP	**Reserved**
			Schmitter Trigger Control
3		MSoRSTB_SMT	0:Disable 1:Enable
2		MSoRSTB_Ro	**10K resistot control**
1		MSoRSTB_R1	**50K resistor control**
0		MSoRSTB_PUPD	**pull-up(0)/pull-down(1) control**

10005D20 MSDCo_CTRL6 MSDC o Pad Control Register 000A

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSoPAD_BACKUP						MSoPAD_RDSEL						MSoPAD_TDSEL			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
15:10		MSoPAD_BACK	**Reserved**

Bit(s)	Mnemonic	Name	Description
		UP	
9:4		MSoPAD_RDSEL	**BIAS PAD TUNE Control**
3:0		MSoPAD_TDSEL	**BIAS PAD TUNE Control**

10005D30 MSDC1_CTRL0 MSDC 1 CLK Pad Control Register 0311

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1CK_BACKUP1				MS1CK_SMT	MS1CK_R0	MS1CK_R1	MS1CK_PUPD	MS1CK_BACKUP0			MS1CK_IES	MS1CK_SR	MS1CK_E8	MS1CK_E4	MS1CK_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MS1CK_BACKUP1	Reserved
11		MS1CK_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS1CK_R0	**10K resistot control**
9		MS1CK_R1	**50K resistor control**
8		MS1CK_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MS1CK_BACKUP0	**Reserved**
4		MS1CK_IES	**Input enable control**
3		MS1CK_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS1CK_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005D40 MSDC1_CTRL1 MSDC 1 CMD Pad Control Register 0211

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1CMD_BACKUP1				MS1C	MS1C	MS1C	MS1C	MS1CMD_BACKUP0			MS1C	MS1C	MS1C	MS1C	MS1C

10005D40 **MSDC1_CTRL1** **MSDC 1 CMD Pad Control Register** **0211**

					MD_SMT	MD_R0	MD_R1	MD_PUPD				MD_IES	MD_SR	MD_E8	MD_E4	MD_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MS1CMD_BACKUP1	Reserved
11		MS1CMD_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS1CMD_R0	**10K resistot control**
9		MS1CMD_R1	**50K resistor control**
8		MS1CMD_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MS1CMD_BACKUP0	**Reserved**
4		MS1CMD_IES	**Input enable control**
3		MS1CMD_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS1CMD_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005D50 **MSDC1_CTRL2** **MSDC 1 DATA Pad Control Register 0** **0011**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1DAT_BACKUP0											MS1DAT_IES	MS1DAT_SR	MS1DAT_E8	MS1DAT_E4	MS1DAT_E2
Type	RW											RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:5		MS1DAT_BACKUP0	**Reserved**

Bit(s)	Mnemonic	Name	Description
4		MS1DAT_IES	**Input enable control**
3		MS1DAT_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS1DAT_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005D60 **MSDC1_CTRL3** **MSDC 1 DATA Pad Control Register 1** **2222**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1DAT3_SMT	MS1DAT3_R0	MS1DAT3_R1	MS1DAT3_PUPD	MS1DAT2_SMT	MS1DAT2_R0	MS1DAT2_R1	MS1DAT2_PUPD	MS1DAT1_SMT	MS1DAT1_R0	MS1DAT1_R1	MS1DAT1_PUPD	MS1DAT0_SMT	MS1DAT0_R0	MS1DAT0_R1	MS1DAT0_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
15		MS1DAT3_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
14		MS1DAT3_R0	**10K resistot control**
13		MS1DAT3_R1	**50K resistor control**
12		MS1DAT3_PUPD	**pull-up(0)/pull-down(1) control**
11		MS1DAT2_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS1DAT2_R0	**10K resistot control**
9		MS1DAT2_R1	**50K resistor control**
8		MS1DAT2_PUPD	**pull-up(0)/pull-down(1) control**
7		MS1DAT1_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
6		MS1DAT1_R0	**10K resistot control**
5		MS1DAT1_R1	**50K resistor control**
4		MS1DAT1_PUPD	**pull-up(0)/pull-down(1) control**
3		MS1DAT0_SMT	**Schmitter Trigger Control**

Bit(s)	Mnemonic	Name	Description
			0:Disable 1:Enable
2		MS1DAT0_R0	**10K resistot control**
1		MS1DAT0_R1	**50K resistor control**
0		MS1DAT0_PUPD	**pull-up(0)/pull-down(1) control**

10005D70 MSDC1_CTRL4 MSDC 1 DATA Pad Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1DAT_BACKUP															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		MS1DAT_BACKUP	**Reserved**

10005D80 MSDC1_CTRL5 MSDC 1 Pad Control Register 00CA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS1PAD_BACKUP						MS1PAD_RDSEL						MS1PAD_TDSEL			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	1	1	0	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
15:10		MS1PAD_BACKUP	**Reserved**
9:4		MS1PAD_RDSEL	**BIAS PAD TUNE Control**
3:0		MS1PAD_TDSEL	**BIAS PAD TUNE Control**

10005D90 MSDC2_CTRL0 MSDC 2 CLK Pad Control Register 0411

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS2CK_BACKUP1				MS2CK_SMT	MS2CK_R0	MS2CK_R1	MS2CK_PUPD	MS2CK_BACKUP0			MS2CK_IES	MS2CK_SR	MS2CK_E8	MS2CK_E4	MS2CK_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MS2CK_BACKUP1	**Reserved**
11		MS2CK_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS2CK_R0	**10K resistot control**
9		MS2CK_R1	**50K resistor control**
8		MS2CK_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MS2CK_BACKUP0	**Reserved**
4		MS2CK_IES	**Input enable control**
3		MS2CK_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS2CK_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005DA0 **MSDC2_CTRL1** **MSDC 2 CMD Pad Control Register** **0511**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS2CMD_BACKUP1				MS2CMD_SMT	MS2CMD_R0	MS2CMD_R1	MS2CMD_PUPD	MS2CMD_BACKUP0			MS2CMD_IES	MS2CMD_SR	MS2CMD_E8	MS2CMD_E4	MS2CMD_E2
Type	RW				RW	RW	RW	RW	RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:12		MS2CMD_BACKUP1	**Reserved**
11		MS2CMD_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS2CMD_R0	**10K resistot control**
9		MS2CMD_R1	**50K resistor control**
8		MS2CMD_PUPD	**pull-up(0)/pull-down(1) control**
7:5		MS2CMD_BACKUP0	**Reserved**

Bit(s)	Mnemonic	Name	Description
4		MS2CMD_IES	**Input enable control**
3		MS2CMD_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS2CMD_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005DB0 MSDC2_CTRL2 **MSDC 2 DATA Pad Control Register 0** **0011**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS2DAT_BACKUP0											MS2DAT_IES	MS2DAT_SR	MS2DAT_E8	MS2DAT_E4	MS2DAT_E2
Type	RW											RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:5		MS2DAT_BACKUP0	**Reserved**
4		MS2DAT_IES	**Input enable control**
3		MS2DAT_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS2DAT_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005DC0 MSDC2_CTRL3 **MSDC 2 DATA Pad Control Register 1** **5555**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS	MS

10005DC0 **MSDC2_CTRL3** **MSDC 2 DATA Pad Control Register 1** **5555**

Name	2DAT3_SMT	2DAT3_R0	2DAT3_R1	2DAT3_PUPD	2DAT2_SMT	2DAT2_R0	2DAT2_R1	2DAT2_PUPD	2DAT1_SMT	2DAT1_R0	2DAT1_R1	2DAT1_PUPD	2DAT0_SMT	2DAT0_R0	2DAT0_R1	2DAT0_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15		MS2DAT3_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
14		MS2DAT3_R0	**10K resistot control**
13		MS2DAT3_R1	**50K resistor control**
12		MS2DAT3_PUPD	**pull-up(0)/pull-down(1) control**
11		MS2DAT2_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
10		MS2DAT2_R0	**10K resistot control**
9		MS2DAT2_R1	**50K resistor control**
8		MS2DAT2_PUPD	**pull-up(0)/pull-down(1) control**
7		MS2DAT1_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
6		MS2DAT1_R0	**10K resistot control**
5		MS2DAT1_R1	**50K resistor control**
4		MS2DAT1_PUPD	**pull-up(0)/pull-down(1) control**
3		MS2DAT0_SMT	**Schmitter Trigger Control** 0:Disable 1:Enable
2		MS2DAT0_R0	**10K resistot control**
1		MS2DAT0_R1	**50K resistor control**
0		MS2DAT0_PUPD	**pull-up(0)/pull-down(1) control**

10005DD0 **MSDC2_CTRL4** **MSDC 2 DATA Pad Control Register 2** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS2DAT_BACKUP															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		MS2DAT_BACK	**Reserved**

Bit(s)	Mnemonic	Name	Description
		UP	

10005DE0 MSDC2_CTRL5 MSDC 2 Pad Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS2PAD_BACKUP						MS2PAD_RDSEL						MS2PAD_TDSEL			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		MS2PAD_BACKUP	Reserved
9:4		MS2PAD_RDSEL	BIAS PAD TUNE Control
3:0		MS2PAD_TDSEL	BIAS PAD TUNE Control

10005DF0 GPIO_TM GPIO DIR Status Selection Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																TM_DIR
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0		TM_DIR	**select gpio_dir or gpio_padoe for register read in address GPIO_DIR 1~9** 0: select gpio_dir for register read 1: select gpio_padoe for register read

10005E00 GPIO_USB USB IDDIG GPIO PULLUP Control Register 0069

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										AGPIO_SEL	MODE18V	MS_BS_PRE_SEL	SELSD	USB20_GPIO49_IDPULLUP	USB20_GPIO34_IDPULLUP	USB20_GPIO38_IDPULLUP
Type										RW	RW	RW	RW	RW	RW	RW
Reset										1	1	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
6		AGPIO_SEL	Reserved
5		MODE18V	Reserved
4		MS_BS_PRE_SEL	Reserved
3		SELSD	**PAD_IDDIG pupd by DA_USB signal when usb_mode is set**
2		USB20_GPIO49_IDPULLUP	1: by DA_USB signal 0: by register
1		USB20_GPIO34_IDPULLUP	**PAD_IDDIG pupd by DA_USB signal when usb_mode is set** 1: by DA_USB signal 0: by register
0		USB20_GPIO38_IDPULLUP	**PAD_IDDIG pupd by DA_USB signal when usb_mode is set** 1: by DA_USB signal 0: by register

10005E10 OD33_CTRL0 18OD33 IO Group TDSEL/RDSEL Control Register 0 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP0						OD33_RDSEL0						OD33_TDSEL0			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP0	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL0	**RDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**
3:0		OD33_TDSEL0	**TDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**

10005E20 OD33_CTRL1 18OD33 IO Group TDSEL/RDSEL Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP1						OD33_RDSEL1						OD33_TDSEL1			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP1	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL1	**RDSEL control register for eint12_tdsel**

Bit(s)	Mnemonic	Name	Description
3:0		OD33_TDSEL1	**TDSEL control register for eint12_tdsel**

10005E30 OD33_CTRL2 18OD33 IO Group TDSEL/RDSEL Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP2						OD33_RDSEL2						OD33_TDSEL2			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP2	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL2	**RDSEL control register for eint17_tdsel**
3:0		OD33_TDSEL2	**TDSEL control register for eint17_tdsel**

10005E40 OD33_CTRL3 18OD33 IO Group TDSEL/RDSEL Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OD33_BACKUP3						OD33_RDSEL3						OD33_TDSEL3			
Type	RW						RW						RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:10		OD33_BACKUP3	**OD33 Control Reserve Registers**
9:4		OD33_RDSEL3	**RDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**
3:0		OD33_TDSEL3	**TDSEL control register for uart2_tdsel, pcm_tdsel, eint0_tdsel and eint5_tdsel**

10005E50 KPAD_CTRL0 Keypad ROW Pad R0/R1/PUPD Control Register 0 0555

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	KPAD_CTRL0_BACKUP2					ROW2_PUPD	ROW2_R1R0		KPAD_CTRL0_BACKUP1	ROW1_PUPD	ROW1_R1R0		KPAD_CTRL0_BACKUP0	ROW0_PUPD	ROW0_R1R0	
Type	RW					RW	RW		RW	RW	RW		RW	RW	RW	
Reset	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15:11		KPAD_CTRL0_BACKUP2	**Reserved**
10		ROW2_PUPD	**pull-up(0)/pull-down(1) control**
9:8		ROW2_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
7		KPAD_CTRL0_BACKUP1	**Reserved**
6		ROW1_PUPD	**pull-up(0)/pull-down(1) control**
5:4		ROW1_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
3		KPAD_CTRL0_BACKUP0	**Reserved**
2		ROW0_PUPD	**pull-up(0)/pull-down(1) control**
1:0		ROW0_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K

10005E60 **KPAD_CTRL1** **Keypad COL Pad R0/R1/PUPD Control Register 1** **0551**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	KPAD_CTRL1_BACKUP2					COL2_PUPD	COL2_R1R0		KPAD_CTRL1_BACKUP1	COL1_PUPD	COL1_R1R0		KPAD_CTRL1_BACKUP0	COL0_PUPD	COL0_R1R0	
Type	RW					RW	RW		RW	RW	RW		RW	RW	RW	
Reset	0	0	0	0	0	1	0	1	0	1	0	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:11		KPAD_CTRL1_BACKUP2	**Reserved**
10		COL2_PUPD	**pull-up(0)/pull-down(1) control**
9:8		COL2_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
7		KPAD_CTRL1_BACKUP1	**Reserved**

Bit(s)	Mnemonic	Name	Description
6		COL1_PUPD	**pull-up(0)/pull-down(1) control**
5:4		COL1_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
3		KPAD_CTRL1_BACKUP0	**Reserved**
2		COL0_PUPD	**pull-up(0)/pull-down(1) control**
1:0		COL0_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K

10005E70 **EINT_CTRL0** **EINT Pad R0/R1/PUPD Control Register 0** **1115**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_CTRL0_BACKUP3	EINT17_PUPD	EINT17_R1R0		EINT_CTRL0_BACKUP2	EINT16_PUPD	EINT16_R1R0		EINT_CTRL0_BACKUP1	EINT15_PUPD	EINT15_R1R0		EINT_CTRL0_BACKUP0	EINT14_PUPD	EINT14_R1R0	
Type	RW	RW	RW		RW	RW	RW		RW	RW	RW		RW	RW	RW	
Reset	0	0	0	1	0	0	0	1	0	0	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15		EINT_CTRL0_BACKUP3	**Reserved**
14		EINT17_PUPD	**pull-up(0)/pull-down(1) control**
13:12		EINT17_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
11		EINT_CTRL0_BACKUP2	**Reserved**
10		EINT16_PUPD	**pull-up(0)/pull-down(1) control**
9:8		EINT16_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K
7		EINT_CTRL0_BACKUP1	**Reserved**
6		EINT15_PUPD	**pull-up(0)/pull-down(1) control**
5:4		EINT15_R1R0	00:High Z 01: 10K

Bit(s)	Mnemonic	Name	Description
			10: 50K 11:10K//50K
3		EINT_CTRL0_BACKUP0	**Reserved**
2		EINT14_PUPD	**pull-up(0)/pull-down(1) control**
1:0		EINT14_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K

10005E80 **EINT_CTRL1** **EINT Pad R0/R1/PUPD Control Register 1** **0001**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_CTRL1_BACKUP0													EINT21_PUPD	EINT21_R1R0	
Type	RW													RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:3		EINT_CTRL1_BACKUP0	**Reserved**
2		EINT21_PUPD	**pull-up(0)/pull-down(1) control**
1:0		EINT21_R1R0	00:High Z 01: 10K 10: 50K 11:10K//50K

10005EB0 **BIAS_CTRL0** **18OD33 IO Group BIAS Control Register 0** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	I2S_BIAS2_CTRL				I2S_BIAS1_CTRL				GATE_MC2_B_CTRL				GATE_MS2_CTRL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:12		I2S_BIAS2_CTRL	**BIAS PAD TUNE Control**
11:8		I2S_BIAS1_CTRL	**BIAS PAD TUNE Control**
7:4		GATE_MC2_B_CTRL	**BIAS PAD TUNE Control**

Bit(s)	Mnemonic	Name	Description
3:0		GATE_MS2_CTRL	BIAS PAD TUNE Control

10005EC0 BIAS_CTRL1 18OD33 IO Group BIAS Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPDIF_BIAS1B_CTRL				SPDIF_BIAS1_CTRL				I2S_BIAS4_CTRL				I2S_BIAS3_CTRL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:12		SPDIF_BIAS1B_CTRL	BIAS PAD TUNE Control
11:8		SPDIF_BIAS1_CTRL	BIAS PAD TUNE Control
7:4		I2S_BIAS4_CTRL	BIAS PAD TUNE Control
3:0		I2S_BIAS3_CTRL	BIAS PAD TUNE Control

10005ED0 BIAS_CTRL2 18OD33 IO Group BIAS Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GATE_DPIB_CTRL				GATE_DPI_CTRL				U_RGM2_BIASB_CTRL				U_RGM2_BIAS_CTRL			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:12		GATE_DPIB_CTRL	BIAS PAD TUNE Control
11:8		GATE_DPI_CTRL	BIAS PAD TUNE Control
7:4		U_RGM2_BIASB_CTRL	BIAS PAD TUNE Control
3:0		U_RGM2_BIAS_CTRL	BIAS PAD TUNE Control

10005F00 DRV_SEL10 GPIO Driving Control Register 10 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL10															

10005F00 DRV_SEL10 GPIO Driving Control Register 10 0000

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL10	Driving strength control, refer to Table1-3 for detail

10005F10 DRV_SEL11 GPIO Driving Control Register 11 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL11															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL11	Driving strength control, refer to Table1-3 for detail

10005F30 DRV_SEL12 GPIO Driving Control Register 12 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL12															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL12	Driving strength control, refer to Table1-3 for detail

10005F40 MSDC3_CTRL3 MSDC 3 DATA Pad Control Register 1 4444

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3DAT3_SMT	MS3DAT3_Ro	MS3DAT3_R1	MS3DAT3_PUPD	MS3DAT2_SMT	MS3DAT2_Ro	MS3DAT2_R1	MS3DAT2_PUPD	MS3DAT1_SMT	MS3DAT1_Ro	MS3DAT1_R1	MS3DAT1_PUPD	MS3DAT0_SMT	MS3DAT0_Ro	MS3DAT0_R1	MS3DAT0_PUPD
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15		MS3DAT3_SMT	Schmitter Trigger Control 0:Disable 1:Enable

Bit(s)	Mnemonic	Name	Description
14		MS3DAT3_R0	10K resistot control
13		MS3DAT3_R1	50K resistor control
12		MS3DAT3_PUPD	pull-up(0)/pull-down(1) control
11		MS3DAT2_SMT	Schmitter Trigger Control 0:Disable 1:Enable
10		MS3DAT2_R0	10K resistot control
9		MS3DAT2_R1	50K resistor control
8		MS3DAT2_PUPD	pull-up(0)/pull-down(1) control
7		MS3DAT1_SMT	Schmitter Trigger Control 0:Disable 1:Enable
6		MS3DAT1_R0	10K resistot control
5		MS3DAT1_R1	50K resistor control
4		MS3DAT1_PUPD	pull-up(0)/pull-down(1) control
3		MS3DAT0_SMT	Schmitter Trigger Control 0:Disable 1:Enable
2		MS3DAT0_R0	10K resistot control
1		MS3DAT0_R1	50K resistor control
0		MS3DAT0_PUPD	pull-up(0)/pull-down(1) control

10005F50 DRV_SEL0 GPIO Driving Control Register 0 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL0	Driving strength control, refer to Table1-3 for detail

10005F60 DRV_SEL1 GPIO Driving Control Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL1	Driving strength control, refer to Table1-3 for detail

10005F70 DRV_SEL2 GPIO Driving Control Register 2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL2	Driving strength control, refer to Table1-3 for detail

10005F80 DRV_SEL3 GPIO Driving Control Register 3 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL3															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL3	Driving strength control, refer to Table1-3 for detail

10005F90 DRV_SEL4 GPIO Driving Control Register 4 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL4															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL4	Driving strength control, refer to Table1-3 for detail

10005FA0 DRV_SEL5 GPIO Driving Control Register 5 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL5															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL5	Driving strength control, refer to Table1-3 for detail

10005FB0 DRV_SEL6 GPIO Driving Control Register 6 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL6															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL6	**Driving strength control, refer to Table1-3 for detail**

10005FC0 MSDC3_CTRL2 MSDC 3 DATA Pad Control Register 0 0014

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MS3DAT_BACKUP											MS3DAT_IES	MS3DAT_SR	MS3DAT_E8	MS3DAT_E4	MS3DAT_E2
Type	RW											RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description
15:5		MS3DAT_BACKUP	**Reserved**
4		MS3DAT_IES	**Input enable control**
3		MS3DAT_SR	**Output Slew Rate Control.** 1: slower slew 0: no slew rate controlled
2:0		MS3DAT_DRV	**Driving Strength Control** 000 : 2mA 001: 4mA 010: 6mA 011: 8mA 100: 10mA 101: 12mA 110: 14mA 111: 16mA

10005FD0 DRV_SEL8 GPIO Driving Control Register 8 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL8															
Type	RW															

10005FD0 **DRV_SEL8** **GPIO Driving Control Register 8** 0000

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL8	Driving strength control, refer to Table1-3 for detail

10005FE0 **DRV_SEL7** **GPIO Driving Control Register 7** 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL7															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL7	Driving strength control, refer to Table1-3 for detail

10005FF0 **DRV_SEL9** **GPIO Driving Control Register 9** 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DRV_SEL9															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DRV_SEL9	Driving strength control, refer to Table1-3 for detail

2 Top Clock Generator

2.1 Introduction

This chapter introduces the top clock generator (TOPCKGEN) and the clock architecture

2.2 Feature list

TOPCKGEN is responsible for generating the following clock signals:

- Free clock generation for whole chip
- Cortex-A7 CPU clock
- Infrastructure and peripheral system clock, including the top level AXI fabric clock
- Multimedia system clock
- Pad clocks to be synchronized with one of the above system

The module TOPCKGEN provides clock source selection. Each clock has several clock source selections and can be turned off as well. When switching certain clock from frequency A to frequency B, make sure frequency A and B are available.

It comprises glitch-free clock MUX and digital clock divider to generate various clock frequencies.

2.3 Block Diagram

2.3.1 Clock Architecture

The clock generator is not only in the top level but also in every partition / system. Below is the location of the top level clock generator

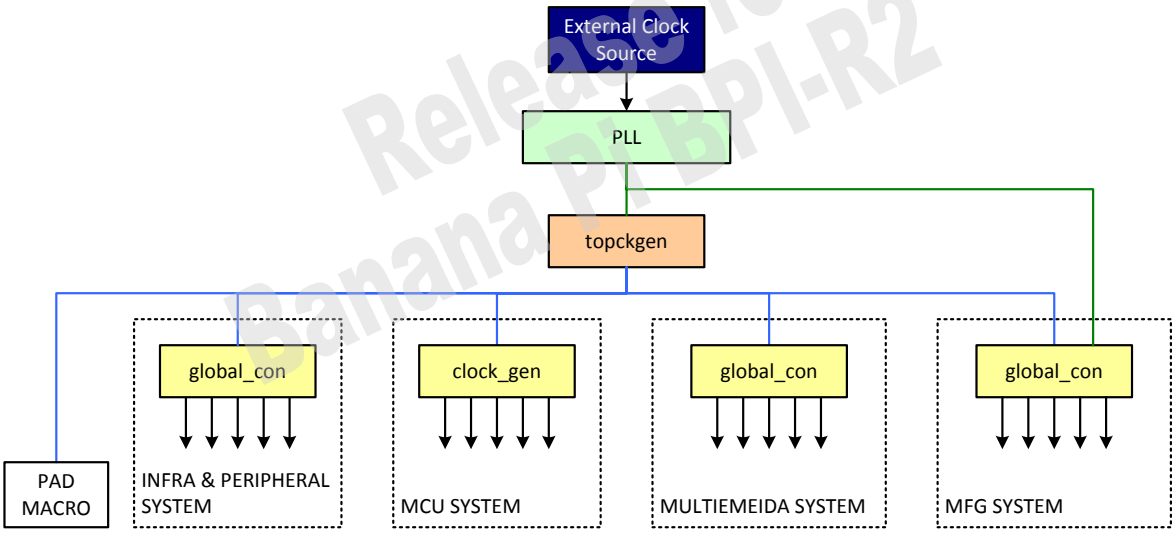


Figure 2-1: Block diagram of clock architecture

2.3.2 Clock Multiplixer

Clock selection and generation have similar structure. Several clock sources are provided. Choose on by specified register setting. The turn-off bit is provided as well to stop the clock output.

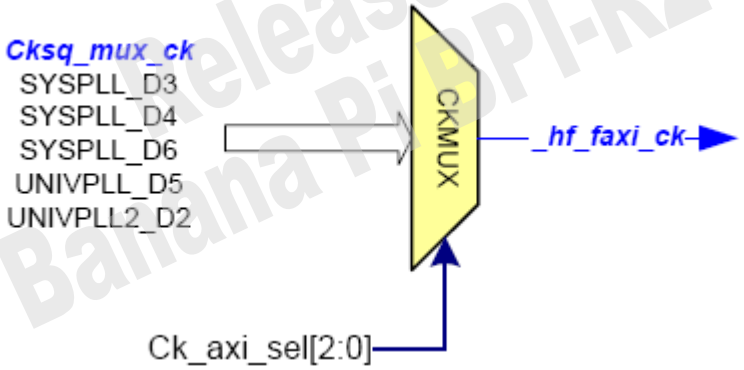


Figure 2-2: Example of Clock Multiplixer

2.4 Clock PLL

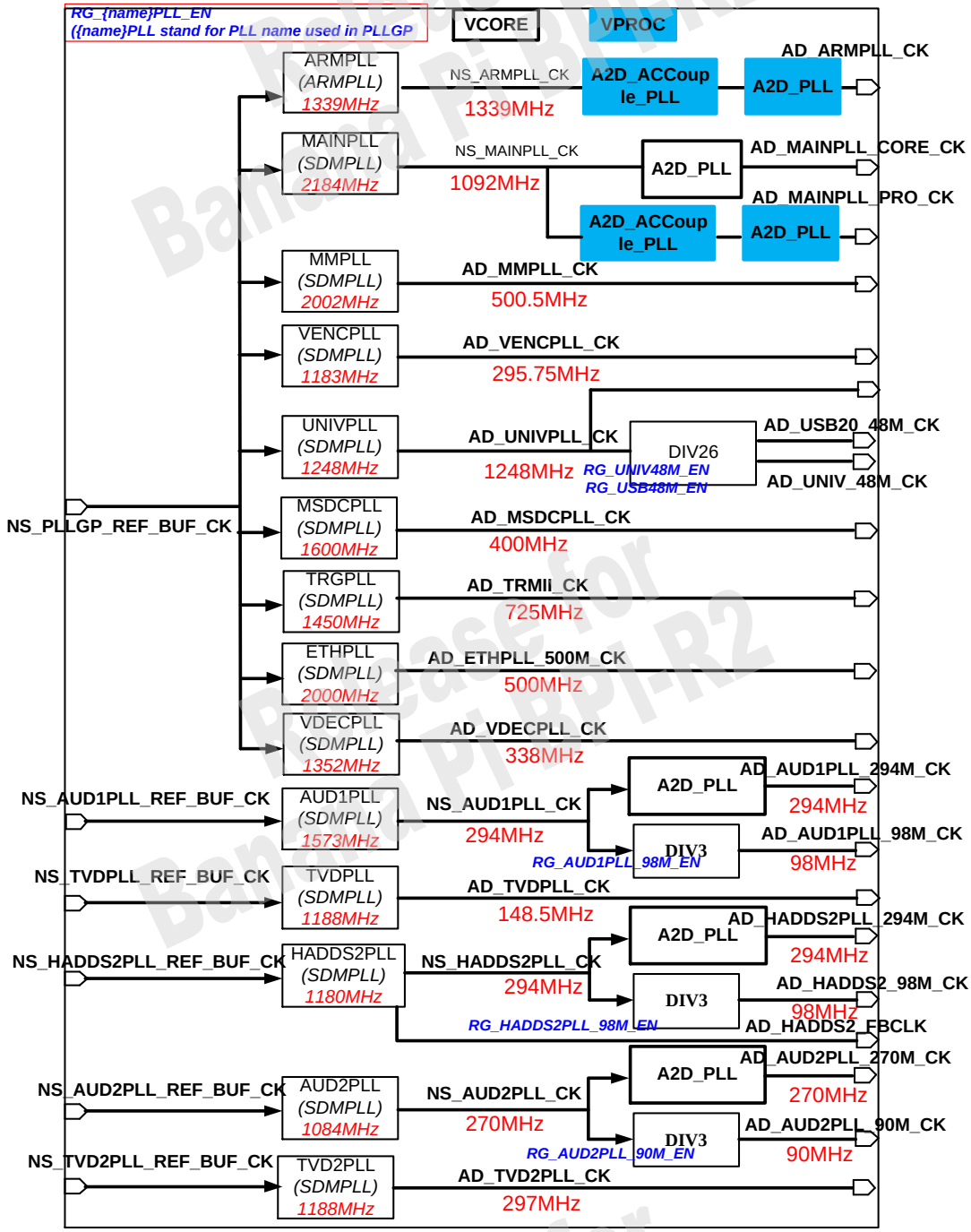


Figure 2-3: PLL block diagram

2.5 PLL Related Control

The following table lists all PLLs inside the application system.

The enabling of PLL can be switched between software control and hardware control. The hardware control is from SCPSYS.

The hopping and SSC features can be switched between software control and hardware control. The hardware control is from FHCTL

Table 2-1: PLL related control

PLL	PLL capability	ABB	FHCTL	SCPSYS	Main usage
AMRPLL	Hopping, SSC	Y	Y	Y	MCU
MAINPLL	Hopping, SSC	Y	Y	Y (backup)	Command bus, peri.
MMPLL	Hopping, SSC	Y	Y	-	3D
VENCPLL	SSC	Y	Y (SSC)	-	Multi-media
UNIVPLL	Fix	Y	-	Y (backup)	USB
MSDCPLL	Hopping, SSC	Y	Y	-	MSDC
MEMPLL (DDRPHY)	Hopping, SSC	-	-	Y	Memory bus, EMI, DDRPHY
MIPIPLL (MIPI)	Hopping, SSC	-	-	-	Display
USB_PHYA (USB)	Fix	-	-	-	USB PHY
TRGPLL	SSC	-	-	-	ETHIFSYS
ETHPLL	SSC	-	-	-	ETHIFSYS
VDECPLL	SSC	-	-	-	VDEC
TVDPLL	SSC	-	-	-	TVE
AUD1PLL	Fix	-	-	-	AUDIO
AUD2PLL	Fix	-	-	-	AUDIO
TVD2PLL	SSC	-	-	-	NR TVE
HADDS2PLL	Fix	-	-	-	AUDIO HDMI

2.6 Clock Gating

The clock gating for module TOPCKGEN is listed in the table below where DCM and turn-off settings are provided.

Table 2-2: Clock gating settings

Register name	Bit	Default	Function name	Description
CLK_MODE	8	1'b0	pdn_md_32k	Turns off 32K clock source to MD

Register name	Bit	Default	Function name	Description
	10	1'b0	pdn_conn_32k	Turns off 32K clock source to CONN
DCM_CFG	[7]	1'b0	dcm_enable	Enables hf_faxi_ck DCM
CLK_SCP_CFG_0	[0]	1'b0	sc_26ck_off_en	Turns on scpsys control path to gate 26MHz
	[1]	1'b0	sc_mem_ck_off_en	Turns on scpsys control path to gate DDRPHY
	[2]	1'b0	sc_axick_off_en	Turns on scpsys control path to gate hf_faxi_ck
	[4]	1'b0	sc_armck_off_en	Turns on scpsys control path to gate hf_farm_ck
	[5]	1'b0	sc_md_32k_off_en	Turns on scpsys control path to gate MD 32kHz
	[7]	1'b0	sc_conn_32k_off_en	Turns on scpsys control path to gate CONN 32kHz
	[9]	1'b0	sc_mac_26m_off_en	Turns on scpsys control path to gate MIPI 26MHz
CLK_SCP_CFG_1	[0]	1'b0	sc_axi_26m_sel_en	Turns on scpsys control path to switch hf_faxi_ck to 26MHz
	[4]	1'b0	sc_axick_dcm_dis_en	Turns on scpsys control path to disable DCM of hf_faxi_ck

2.7 Frequency Meter

There are two frequency meters inside TOPCKGEN. One is for PLLs and TEST clock called abist_fmeter. The other is for clocks generated from TOPCKGEN called ckgen_fmeter.

Both structures have PAD output that can observe frequency directly instead of reading results from the frequency meter. Abist_fmeter is outputted to CLKM[0], and ckgen_fmeter is outputted to DEBUG_MON[2].

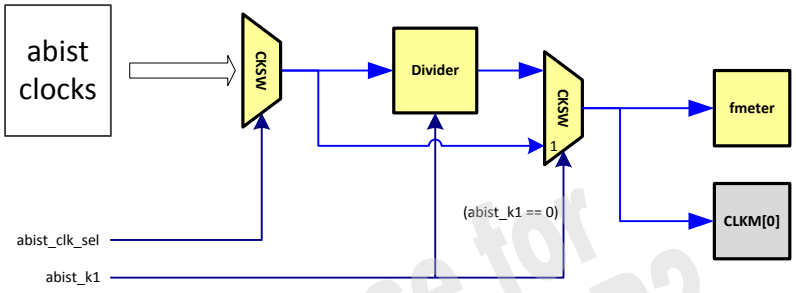


Figure 2-4: ABIST FMETER structure

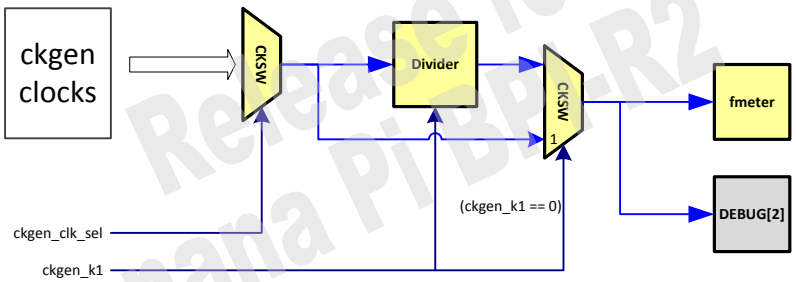


Figure 2-5: CKGEN FMETER structure

2.8 Register Definition

Module name: TOPCKGEN Base address: (+10000000h)

Address	Name	Width	Register Function
10000000	CLK_MODE	32	Clock 26M, 32K PDN Control Register
10000004	DCM_CFG	32	AXI Bus Clock DCM Control Register
10000040	CLK_CFG_0	32	Function Clock Selection Register 0
10000044	CLK_CFG_0_SET	32	SET control of CLK_CFG_0
10000048	CLK_CFG_0_CLR	32	CLR control of CLK_CFG_0
10000050	CLK_CFG_1	32	Function clock selection register 1
10000054	CLK_CFG_1_SET	32	SET control of CLK_CFG_1
10000058	CLK_CFG_1_CLR	32	CLR control of CLK_CFG_1
10000060	CLK_CFG_2	32	Function clock selection register 2
10000064	CLK_CFG_2_SET	32	SET control of CLK_CFG_2
10000068	CLK_CFG_2_CLR	32	CLR control of CLK_CFG_2
10000070	CLK_CFG_3	32	Function clock selection register 3
10000074	CLK_CFG_3_SET	32	SET control of CLK_CFG_3
10000078	CLK_CFG_3_CLR	32	CLR control of CLK_CFG_3
10000080	CLK_CFG_4	32	Function clock selection register 4
10000084	CLK_CFG_4_SET	32	SET control of CLK_CFG_4
10000088	CLK_CFG_4_CLR	32	CLR control of CLK_CFG_4
10000090	CLK_CFG_5	32	Function clock selection register 5
10000094	CLK_CFG_5_SET	32	SET control of CLK_CFG_5
10000098	CLK_CFG_5_CLR	32	CLR control of CLK_CFG_5
100000A0	CLK_CFG_6	32	Function clock selection register 6
100000A4	CLK_CFG_6_SET	32	SET control of CLK_CFG_6

100000A8	CLK_CFG_6_CLR	32	CLR control of CLK_CFG_6
100000B0	CLK_CFG_7	32	Function clock selection register 7
100000B4	CLK_CFG_7_SET	32	SET control of CLK_CFG_7
100000B8	CLK_CFG_7_CLR	32	CLR control of CLK_CFG_7
100000C0	CLK_CFG_12	32	Function clock selection register 12
100000C4	CLK_CFG_12_SET	32	SET control of CLK_CFG_12
100000C8	CLK_CFG_12_CLR	32	CLR control of CLK_CFG_12
100000D0	CLK_CFG_13	32	Function clock selection register 13
100000D4	CLK_CFG_13_SET	32	SET control of CLK_CFG_13
100000D8	CLK_CFG_13_CLR	32	CLR control of CLK_CFG_13
100000E0	CLK_CFG_14	32	Function clock selection register 14
100000E4	CLK_CFG_14_SET	32	SET control of CLK_CFG_14
100000E8	CLK_CFG_14_CLR	32	CLR control of CLK_CFG_14
100000F0	CLK_CFG_15	32	Function clock selection register 15
100000F4	CLK_CFG_15_SET	32	SET control of CLK_CFG_15
100000F8	CLK_CFG_15_CLR	32	CLR control of CLK_CFG_15
10000100	CLK_CFG_8	32	Function clock selection register 8
10000104	CLK_CFG_9	32	Function clock selection register 9
10000108	CLK_CFG_10	32	Debug monitor clock selection register
1000010C	CLK_CFG_11	32	Debug monitor divider control register
10000120	CLK_AUDDIV_0	32	audio clock divider control register0
10000124	CLK_AUDDIV_1	32	audio clock divider control register1
10000128	CLK_AUDDIV_2	32	audio clock divider control register2
1000012C	CLK_AUDDIV_3	32	audio clock divider control register3
10000150	CLK_8BDAC_CFG	32	8bdac clock divider control register
10000200	CLK_SCP_CFG_0	32	SCP control register 0
10000204	CLK_SCP_CFG_1	32	SCP control register 1
10000210	CLK_MISC_CFG_0	32	Internal clk_rtc divider control register
10000214	CLK_MISC_CFG_1	32	Frequency meter divider control register
10000220	CLK26CALI_0	32	Frequency meter control register 0
10000224	CLK26CALI_1	32	Frequency meter control register 1
10000228	CLK26CALI_2	32	Frequency meter control register 2
1000022C	CKSTA_REG	32	Function clock selection status register
10000230	TEST_MODE_C	32	Test mode control register

	FG		
1000030C	MBIST_CFG_1	32	Debug monitor selection register 1
10000310	RESET_DEGLITCH_KEY	32	Reset deglitch enable key register
10000314	MBIST_CFG_3	32	Debug monitor selection register 3
10000318	BOOT_TRAP	32	Boot strap register

10000000 CLK_MODE Clock 26M, 32K PDN Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						pdn_conn_32k		pdn_md_32k								topckgen_en
Type						RW		RW								RW
Reset						0		0								0

Bit(s)	Name	Description
10	pdn_conn_32k	**Turns off 32K clock source to CONN** Turn off this clock in flight mode 1: Enable turn-off
8	pdn_md_32k	**Turns off 32K clock source to MD** Turn off this clock in flight mode 1: Enable turn-off
0	topckgen_en	**Enables TOPCKGEN** 0: Enable 1: Disable

10000004 DCM_CFG AXI Bus Clock DCM Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dcm_dbc_enable	dcm_dbc_cnt							dcm_enable			dcm_full_fsel				
Type	RW	RW							RW			RW				
Reset	0	0	0	0	0	0	0	0	0			0	0	0	0	0

Bit(s)	Name	Description
15	dcm_dbc_enable	**Enables DCM de-bounce counter** 1: Enable DCM de-bounce counter
14:8	dcm_dbc_cnt	**DCM de-bounce counter**
7	dcm_enable	**Enables hf_faxi_ck DCM** 1: Enable DCM
4:0	dcm_full_fsel	**Selects hf_faxi_ck DCM clock**

Bit(s)	Name	Description
		1xxxx: hd_faxi_ck = hf_faxi_ck 01xxx: hd_faxi_ck = hf_faxi_ck/2 001xx: hd_faxi_ck = hf_faxi_ck/4 0001x: hd_faxi_ck = hf_faxi_ck/8 00001: hd_faxi_ck =hf_faxi_ck/16 00000: hd_faxi_ck = hf_faxi_ck/32

10000040 CLK_CFG_0 Function Clock Selection Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_mm			clk_mm_inv		clk_mm_sel			pdn_ddrphycfg			clk_ddrphycfg_inv				clk_ddrphycfg_sel
Type	RW			RW		RW			RW			RW				RW
Reset	0			0		0	0	0	0			0				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_mem			clk_mem_inv				clk_mem_sel	pdn_axi			clk_axi_inv		clk_axi_sel		
Type	RW			RW				RW	RW			RW		RW		
Reset	0			0				0	0			0		0	0	0

Bit(s)	Name	Description
31	pdn_mm	
28	clk_mm_inv	**Inverts hf_fmm_ck clock phase** 1: Enable phase inversion
26:24	clk_mm_sel	**Selects hf_fmm_ck clock mux** 0: clk26m 1: vencpll_ck 2: syspll1_d2 3: syspll_d5 4: syspll1_d4 5: univpll_d5 6: univpll2_d2 7: dmpll_ck
23	pdn_ddrphycfg	**Turns off hf_fddrphycfg_ck** 1: Enable clock-off
20	clk_ddrphycfg_inv	**Inverts hf_fddrphycfg_ck clock phase** 1: Enable phase inversion
16	clk_ddrphycfg_sel	**Selects hf_fddrphycfg_ck clock mux** 0: clk26m 1: syspll1_d8
15	pdn_mem	**Turns off hf_fmem_ck** 1: Enable clock-off
12	clk_mem_inv	**Inverts hf_fmem_ck clock phase** 1: Enable phase inversion
8	clk_mem_sel	**Selects hf_fmem_ck clock mux** 0: clk26m 1: dmpll_ck
7	pdn_axi	**Turns off hf_faxi_ck** 1: Enable clock-off

Bit(s)	Name	Description
4	clk_axi_inv	**Inverts hf_faxi_ck clock phase** 1: Enable phase inversion
2:0	clk_axi_sel	**Selects hf_faxi_ck clock mux** 0: clk26m 1: syspll1_d2 2: syspll_d5 3: syspll1_d4 4: univpll_d5 5: univpll2_d2 6: dmpll_ck 7: dmpll_d2

10000044 CLK_CFG_0_SET SET control of CLK_CFG_0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_0_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_0_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_0_set	**Sets the correspondent bit of CLG_CFG_SEL_0** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000048 CLK_CFG_0_CLR CLR control of CLK_CFG_0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_0_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_0_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_0_clr	**Clears the correspondent bit of CLG_CFG_SEL_0** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000050 CLK_CFG_1 Function clock selection register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_camt			clk_camtg		clk_camtg_sel			pdn_mfg			clk_mfg_		clk_mfg_sel		

	g			_inv								inv				
Type	RW			RW		RW			RW			RW		RW		
Reset	0			0		0	0	0	0			0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_vdec			clk_vdec_inv	clk_vdec_sel				pdn_pwm			clk_pwm_inv			clk_pwm_sel	
Type	RW			RW	RW				RW			RW			RW	
Reset	0			0	0	0	0	0	0			0			0	0

Bit(s)	Name	Description
31	pdn_camtg	**Turns off hf_fcamtg_ck** 1: Enable clock-off
28	clk_camtg_inv	**Inverts hf_fcamtg_ck clock phase** 1: Enable phase inversion
26:24	clk_camtg_sel	**Selects hf_fcamtg_ck clock mux** 0: clk26m 1: univpll_d26 2: univpll2_d2 3: syspll3_d2 4: syspll3_d4 5: msdcpll_d2 6: mmpll_d2
23	pdn_mfg	**Turns off hf_fmfg_ck** 1: Enable clock-off
20	clk_mfg_inv	**Inverts hf_fmfg_ck clock phase** 1: Enable phase inversion
18:16	clk_mfg_sel	**Selects hf_fmfg_ck clock mux** 0: clk26m 1: mmpll_ck 2: dmpll_x2_ck 3: msdcpll_ck 4: clk26m 5: syspll_d3 6: univpll_d3 7: univpll1_d2
15	pdn_vdec	**turns off hf_fvdec_ck** 1: Enable clock-off
12	clk_vdec_inv	**Inverts hf_fvdec_ck clock phase** 1: Enable phase inversion
11:8	clk_vdec_sel	**Selects hf_fvdec_ck clock mux** 0: clk26m 1: AD_VDECPLL_CK 2: syspll_d5 3: syspll1_d4 4: univpll_d5 5: univpll2_d2 6: AD_VENCPLL_CK 7: msdcpll_d2 8: mmpll_d2
7	pdn_pwm	**Turns off f_fpwm_ck** 1: Enable clock-off
4	clk_pwm_inv	**Inverts f_fpwm_ck clock phase** 1: Enable phase inversion
1:0	clk_pwm_sel	**Selects f_fpwm_ck clock mux** 0: clk26m

Bit(s)	Name	Description
		1: univpll2_d4 2: univpll3_d2 3: univpll1_d4

10000054 CLK_CFG_1_SET SET control of CLK_CFG_1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_1_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_1_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_1_set	**Sets the correspondent bit of CLG_CFG_SEL_1** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000058 CLK_CFG_1_CLR CLR control of CLK_CFG_1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_1_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_1_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_1_clr	**Clears the correspondent bit of CLG_CFG_SEL_1** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000060 CLK_CFG_2 Function clock selection register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_msdc30_0			clk_msdc30_0_inv		clk_msdc30_0_sel			pdn_usb20			clk_usb20_inv			clk_usb20_sel	
Type	RW			RW		RW			RW			RW			RW	
Reset	0			0		0	0	0	0			0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_spi0			clk_spi0_inv		clk_spi0_sel			pdn_uart			clk_uart_inv				clk_uart_sel

Type	RW			RW		RW			RW			RW				RW
Reset	0			0		0	0	0	0			0				0

Bit(s)	Name	Description
31	pdn_msdc30_0	**Turns off hf_fmsdc30_0_ck** 1: Enable clock-off
28	clk_msdc30_0_inv	**Inverts hf_fmsdc30_0_ck clock phase** 1: Enable phase inversion
26:24	clk_msdc30_0_sel	**Selects hf_fmsdc30_0_ck clock mux** 0: clk26m 1: msdcpll_d2 2: syspll2_d2 3: syspll1_d4 4: univpll1_d4 5: univpll2_d4
23	pdn_usb20	**Turns off f_fusb20_ck** 1: Enable clock-off
20	clk_usb20_inv	**Inverts f_fusb20_ck clock phase** 1: Enable phase inversion
17:16	clk_usb20_sel	**Selects f_fusb20_ck clock mux** 0: clk26m 1: univpll1_d8 2: univpll3_d4
15	pdn_spi0	**Turns off hf_fspi0_ck** 1: Enable clock-off
12	clk_spi0_inv	**Inverts hf_fspi0_ck clock phase** 1: Enable phase inversion
10:8	clk_spi0_sel	**Selects hf_fspi0_ck clock mux** 0: clk26m 1: syspll3_d2 2: syspll4_d2 3: univpll2_d4 4: univpll1_d8
7	pdn_uart	**Turns off f_fuart_ck** 1: Enable clock-off
4	clk_uart_inv	**Inverts f_fuart_ck clock phase** 1: Enable phase inversion
0	clk_uart_sel	**Selects f_fuart_ck clock mux** 0: clk26m 1: univpll2_d8

10000064 CLK_CFG_2_SET SET control of CLK_CFG_2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_2_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_2_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_2_set	**Sets the correspondent bit of CLG_CFG_SEL_2** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000068 CLK_CFG_2_CLR CLR control of CLK_CFG_2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_2_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_2_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_2_clr	**Clears the correspondent bit of CLG_CFG_SEL_2** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000070 CLK_CFG_3 Function clock selection register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_aud_intbus			clk_aud_intbus_inv		clk_aud_intbus_sel			pdn_audio			clk_audio_inv				clk_audio_sel
Type	RW			RW		RW			RW			RW				RW
Reset	0			0		0	0	0	0			0				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_msdc30_2			clk_msdc30_2_inv		clk_msdc30_2_sel			pdn_msdc30_1			clk_msdc30_1_inv		clk_msdc30_1_sel		
Type	RW			RW		RW			RW			RW		RW		
Reset	0			0		0	0	0	0			0		0	0	0

Bit(s)	Name	Description
31	pdn_aud_intbus	**Turns off hf_faud_intbus_ck** 1: Enable clock-off
28	clk_aud_intbus_inv	**Inverts hf_faud_intbus_ck clock phase** 1: Enable phase inversion
26:24	clk_aud_intbus_sel	**Selects hf_faud_intbus_ck clock mux** 0: clk26m 1: syspll1_d4 2: syspll3_d2 3: syspll4_d2 4: univpll3_d2 5: univpll2_d4
23	pdn_audio	**Turns off hf_faudio_ck**

Bit(s)	Name	Description
		1: Enable clock-off
20	clk_audio_inv	**Inverts hf_faudio_ck clock phase** 1: Enable phase inversion
16	clk_audio_sel	**Selects hf_faudio_ck clock mux** 0: f_f26m_ck 1: syspll1_d16
15	pdn_msdc30_2	**Turns off hf_fmsdc30_2_ck** 1: Enable clock-off
12	clk_msdc30_2_inv	**Inverts hf_fmsdc30_2_ck clock phase** 1: Enable phase inversion
10:8	clk_msdc30_2_sel	**Selects hf_fmsdc30_2_ck clock mux** 0: clk26m 1: msdcpll_d2 2: syspll2_d2 3: syspll1_d4 4: univpll1_d4 5: univpll2_d4
7	pdn_msdc30_1	**Turns off hf_fmsdc30_1_ck** 1: Enable clock-off
4	clk_msdc30_1_inv	**Inverts hf_fmsdc30_1_ck clock phase** 1: Enable phase inversion
2:0	clk_msdc30_1_sel	**Selects hf_fmsdc30_1_ck clock mux** 0: clk26m 1: msdcpll_d2 2: syspll2_d2 3: syspll1_d4 4: univpll1_d4 5: univpll2_d4

10000074 CLK_CFG_3_SET SET control of CLK_CFG_3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_3_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_3_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_3_set	**Sets the correspondent bit of CLG_CFG_SEL_3** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000078 CLK_CFG_3_CLR CLR control of CLK_CFG_3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_3_clr[31:16]															

Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_3_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_3_clr	**Clears the correspondent bit of CLG_CFG_SEL_3** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000080 CLK_CFG_4 Function clock selection register 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_dpi1			clk_dpi1_inv			clk_dpi1_sel		pdn_dpi0			clk_dpi0_inv		clk_dpi0_sel		
Type	RW			RW			RW		RW			RW		RW		
Reset	0			0			0	0	0			0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_scp			clk_scp_inv			clk_scp_sel		pdn_pmicspi			clk_pmicspi_inv	clk_pmicspi_sel			
Type	RW			RW			RW		RW			RW	RW			
Reset	0			0			0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	pdn_dpi1	**Turns off hf_fdpi1_ck** 1: Enable clock-off
28	clk_dpi1_inv	**Inverts hf_fdpi1_ck clock phase** 1: Enable phase inversion
25:24	clk_dpi1_sel	**Selects hf_fdpi1_ck clock mux** 0: clk26m 1: tvdpll 2: tvdpll_d2 3: tvdpll_d4
23	pdn_dpi0	**Turns off hf_fdpi0_ck** 1: Enable clock-off
20	clk_dpi0_inv	**Inverts hf_fdpi0_ck clock phase** 1: Enable phase inversion
18:16	clk_dpi0_sel	**Selects hf_fdpi0_ck clock mux** 0: clk26m 1: mipipll 2: mipipll_d2 3: mipipll_d4 4: f_f26m_ck 5: tvdpll_ck 6: tvdpll_d2 7: tvdpll_d4
15	pdn_scp	**Turns off hf_fscp_ck** 1: Enable clock-off
12	clk_scp_inv	**Inverts hf_fscp_ck clock phase**

Bit(s)	Name	Description
		1: Enable phase inversion
9:8	clk_scp_sel	**Selects hf_fscp_ck clock mux** 0: clk26m 1: syspll1_d8 2: dmpll_d2 3: dmpll_d4
7:5	pdn_pmicspi	**Turns off hf_fpmicspi_ck** 1: Enable clock-off
4	clk_pmicspi_inv	**Inverts hf_fpmicspi_ck clock phase** 1: Enable phase inversion
3:0	clk_pmicspi_sel	**Selects hf_fpmicspi_ck clock mux** 0: clk26m 1: syspll1_d8 2: syspll2_d4 3: syspll4_d2 4: syspll3_d4 5: syspll2_d8 6: syspll1_d16 7: univpll3_d4 8: univpll_d26 9: dmpll_d2 10: dmpll_d4

10000084 CLK_CFG_4_SET SET control of CLK_CFG_4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_4_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_4_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_4_set	**Sets the correspondent bit of CLG_CFG_SEL_4** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000088 CLK_CFG_4_CLR CLR control of CLK_CFG_4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_4_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_4_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_4_clr	**Clears the correspondent bit of CLG_CFG_SEL_4** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000090 **CLK_CFG_5** **Function clock selection register 5** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_dpilvds			clk_dpilvds_inv		clk_dpilvds_sel			pdn_apll			clk_apll_inv		clk_apll_sel		
Type	RW			RW		RW			RW			RW		RW		
Reset	0			0		0	0	0	0			0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_hdmi			clk_hdmi_inv			clk_hdmi_sel		pdn_tve			clk_tve_inv		clk_tve_sel		
Type	RW			RW			RW		RW			RW		RW		
Reset	0			0			0	0	0			0		0	0	0

Bit(s)	Name	Description
31	pdn_dpilvds	**Turns off hf_fdpilvds_ck** 1: Enable clock-off
28	clk_dpilvds_inv	**Inverts hf_fdpilvds_ck clock phase** 1: Enable phase inversion
26:24	clk_dpilvds_sel	**Selects hf_fdpilvds_ck clock mux** 0: clk26m 1: lvdspll 2: lvdspll_d2 3: lvdspll_d4 4: lvdspll_d8 5: fpc_ck 6: clk26m 7: clk26m
23	pdn_apll	**Turns off f_fapll_ck** 1: Enable clock-off
20	clk_apll_inv	**Inverts f_fapll_ck clock phase** 1: Enable phase inversion
18:16	clk_apll_sel	**Selects f_fapll_ck clock mux** 0: clk26m 1: audpll 2: audpll_d4 3: audpll_d8 4: audpll_d16 5: audpll_d24 6: clk26m 7: clk26m
15	pdn_hdmi	**Turns off hf_fhdmi_ck** 1: Enable clock-off
12	clk_hdmi_inv	**Inverts hf_fhdmi_ck clock phase** 1: Enable phase inversion
9:8	clk_hdmi_sel	**Selects hf_fhdmi_ck clock mux** 0: clk26m 1: hdmipll

Bit(s)	Name	Description
		2: hdmipll_d2 3: hdmipll_d3
7	pdn_tve	**Turns off f_ftve_ck** 1: Enable clock-off
4	clk_tve_inv	**Inverts f_ftve_ck clock phase** 1: Enable phase inversion
2:0	clk_tve_sel	**Selects f_ftve_ck clock mux** 0: clk26m 1: mipipll 2: mipipll_d2 3: mipipll_d4 4: clk_26m 5: tvdpll 6: tvdpll_d2 7: tvdpll_d4

10000094 CLK_CFG_5_SET SET control of CLK_CFG_5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_5_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_5_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_5_set	**Sets the correspondent bit of CLG_CFG_SEL_5** 0: Unchanged 1: Set 1'b1 to the correspondent bit

10000098 CLK_CFG_5_CLR CLR control of CLK_CFG_5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_5_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_5_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_5_clr	**Clears the correspondent bit of CLG_CFG_SEL_5** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000A0 **CLK_CFG_6** **Function clock selection register 6** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_emmc_hclk						clk_emmc_hclk_sel		pdn_eth_50m			clk_eth_50m_inv		clk_eth_50m_sel		
Type	RW						RW		RW			RW		RW		
Reset	0						0	0	0			0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_nfi2x			clk_nfi2x_inv		clk_nfi2x_sel			pdn_rtc			clk_rtc_inv			clk_rtc_sel	
Type	RW			RW		RW			RW			RW			RW	
Reset	0			0		0	0	0	0			0			0	0

Bit(s)	Name	Description
31	pdn_emmc_hclk	**Turns off hf_f_emmc_hclk_ck** 1: Enable clock-off
25:24	clk_emmc_hclk_sel	**Selection hf_f_emmc_hclk_ck** 0: clk26m 1: syspll1_d2 2: syspll1_d4 3: syspll2_d2
23	pdn_eth_50m	**Turns off f_feth_50m_ck** 1: Enable clock-off
20	clk_eth_50m_inv	**Inverts f_feth_50m_ck clock phase** 1: Enable phase inversion
18:16	clk_eth_50m_sel	**Selects f_feth_50m_ck clock mux** 0: clk26m 1: syspll3_d4 2: univpll2_d8 3: lvdspll_eth 4: univpll_d26 5: syspll2_d8 6: syspll4_d4 7: univpll3_d8
15	pdn_nfi2x	**Turns off hf_fnfi2x_ck** 1: Enable clock-off
12	clk_nfi2x_inv	**Inverts hf_fnfi2x_ck clock phase** 1: Enable phase inversion
10:8	clk_nfi2x_sel	**Selects hf_fnfi2x_ck clock mux** 0: clk26m 1: syspll2_d2 2: syspll_d7 3: univpll3_d2 4: syspll2_d4 5: univpll3_d4 6: syspll4_d4 7: clk26m
7	pdn_rtc	**Turns off f_frtc_ck** 1: Enable clock-off
4	clk_rtc_inv	**Inverts f_frtc_ck clock phase** 1: Enable phase inversion
1:0	clk_rtc_sel	**Selects f_frtc_ck clock mux** 0: 32k_internal

Bit(s)	Name	Description
		1: 32k_external 2: clk26m 3: univpll3_d8

100000A4 CLK_CFG_6_SET SET control of CLK_CFG_6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_6_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_6_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_6_set	**Sets the correspondent bit of CLG_CFG_SEL_6** 0: Unchanged 1: Set 1'b1 to the correspondent bit

100000A8 CLK_CFG_6_CLR CLR control of CLK_CFG_6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_6_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_6_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_6_clr	**Clears the correspondent bit of CLG_CFG_SEL_6** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000B0 CLK_CFG_7 Function clock selection register 7 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_osd			clk_osd_inv		clk_osd_sel			pdn_nr			clk_nr_inv		clk_nr_sel		
Type	RW			RW		RW			RW			RW		RW		
Reset	0			0		0	0	0	0			0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_di			clk_di_inv			clk_di_sel		pdn_flash			clk_flash_inv		clk_flash_sel		
Type	RW			RW			RW		RW			RW		RW		
Reset	0			0			0	0	0			0		0	0	0

Bit(s)	Name	Description
31	pdn_osd	**Turns off hf_fosd_ck** 1: Enable clock-off
28	clk_osd_inv	**Inverts hf_fosd_ck clock phase** 1: Enable phase inversion
26:24	clk_osd_sel	**Selects hf_fosd_ck clock mux** 0: clk26m 1: AD_VENCPLL_CK 2: syspll1_d2 3: syspll1_d4 4: univpll_d5 5: univpll1_d2 6: univpll2_d2 7: dmpll_ck
23	pdn_nr	**Turns off f_fnr_ck** 1: Enable clock-off
20	clk_nr_inv	**Inverts f_fnr_ck clock phase** 1: Enable phase inversion
18:16	clk_nr_sel	**Selects f_fnr_ck clock mux** 0: clk26m 1: AD_VENCPLL_CK 2: syspll1_d2 3: syspll1_d4 4: univpll_d5 5: univpll1_d2 6: univpll2_d2 7: dmpll_ck
15	pdn_di	**Turns off hf_fdi_ck** 1: Enable clock-off
12	clk_di_inv	**Inverts hf_fdi_ck clock phase** 1: Enable phase inversion
9:8	clk_di_sel	**Selects hf_fdi_ck clock mux** 0: clk26m 1: tvd2pll_ck 2: tvd2pll_d2 3: clk26m
7	pdn_flash	**Turns off f_fflash_ck** 1: Enable clock-off
4	clk_flash_inv	**Inverts f_fflash_ck clock phase** 1: Enable phase inversion
2:0	clk_flash_sel	**Selects f_fflash_ck clock mux** 0: clk26m_d8 1: clk26m 2: syspll2_d8 3: syspll3_d4 4: univpll3_d4 5: syspll4_d2 6: syspll2_d4 7: univpll2_d4

100000B4 CLK_CFG_7_SET SET control of CLK_CFG_7 00000000

T

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_7_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_7_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_7_set	**Sets the correspondent bit of CLG_CFG_SEL_7** 0: Unchanged 1: Set 1'b1 to the correspondent bit

100000B8 **CLK_CFG_7_CLR** **CLR control of CLK_CFG_7** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_7_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_7_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_7_clr	**Clears the correspondent bit of CLG_CFG_SEL_7** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000C0 **CLK_CFG_12** **Function clock selection register 12** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_asm_m			clk_asm_m_inv			clk_asm_m_sel		pdn_asm_l			clk_asm_l_inv			clk_asm_l_sel	
Type	RW			RW			RW		RW			RW			RW	
Reset	0			0			0	0	0			0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_intdir			clk_intdir_inv			clk_intdir_sel		pdn_hdmirx_bist			clk_hdmirx_bist_inv		clk_hdmirx_bist_sel		
Type	RW			RW			RW		RW			RW		RW		
Reset	0			0			0	0	0			0		0	0	0

Bit(s)	Name	Description
31	pdn_asm_m	**Turns off hf_fasm_m_ck** 1: Enable clock-off
28	clk_asm_m_inv	**Inverts hf_fasm_m_ck clock phase**

Bit(s)	Name	Description
		1: Enable phase inversion
25:24	clk_asm_m_sel	**Selects hf_fasm_m_ck clock mux** 0: clk26m 1: univpll2_d4 2: univpll2_d2 3: syspll_d5
23	pdn_asm_l	**Turns off f_fasm_l_ck** 1: Enable clock-off
20	clk_asm_l_inv	**Inverts f_fasm_l_ck clock phase** 1: Enable phase inversion
17:16	clk_asm_l_sel	**Selects f_fasm_l_ck clock mux** 0: clk26m 1: univpll2_d4 2: univpll2_d2 3: syspll_d5
15	pdn_intdir	**Turns off hf_fint_dir_ck** 1: Enable clock-off
12	clk_intdir_inv	**Inverts hf_fintdir_ck clock phase** 1: Enable phase inversion
9:8	clk_intdir_sel	**Selects hf_fintdir_ck clock mux** 0: clk26m 1: mmpll_ck 2: syspll_d2 3: univpll_d2
7	pdn_hdmirx_bist	**Turns off f_fhdmirx_bist_ck** 1: Enable clock-off
4	clk_hdmirx_bist_inv	**Inverts f_fhdmirx_bist_ck clock phase** 1: Enable phase inversion
2:0	clk_hdmirx_bist_sel	**Selects f_fhdmirx_bist_ck clock mux** 0: clk26m 1: syspll_d3 2: clk26m 3: syspll1_d16 4: syspll4_d2 5: syspll1_d4 6: AD_VENCPLL_CK 7: clk26m

100000C4 **CLK_CFG_12_SET** **SET control of CLK_CFG_12** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_12_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_12_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_12_set	**Sets the correspondent bit of CLG_CFG_SEL_12**

Bit(s)	Name	Description
		0: Unchanged 1: Set 1'b1 to the correspondent bit

100000C8 CLK_CFG_12_CLR CLR control of CLK_CFG_12 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_12_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_12_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_12_clr	**Clears the correspondent bit of CLG_CFG_SEL_12** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000D0 CLK_CFG_13 Function clock selection register 13 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_ethif_m			clk_ethif_inv		clk_ethif_sel			pdn_ms_card					clk_ms_26m_13m_sel	clk_ms_card_sel	
Type	RW			RW		RW			RW					RW	RW	
Reset	0			0		0	0	0	0					0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_host_spi			clk_host_spi_inv		clk_host_spi_sel			pdn_asm_h			clk_asm_h_inv			clk_asm_h_sel	
Type	RW			RW		RW			RW			RW			RW	
Reset	0			0		0	0	0	0			0			0	0

Bit(s)	Name	Description
31	pdn_ethif_m	**Turns off hf_fethif_ck** 1: Enable clock-off
28	clk_ethif_inv	**Inverts hf_fethif_ck clock phase** 1: Enable phase inversion
26:24	clk_ethif_sel	**Selects hf_fethif_ck clock mux** 0: clk26m 1: syspll1_d2 2: syspll_d5 3: syspll1_d4 4: univpll_d5 5: univpll1_d2 6: dmpll_ck 7: dmpll_d2

Bit(s)	Name	Description
23	pdn_ms_card	**Turns off f_fms_card_ck** 1: Enable clock-off
18	clk_ms_26m_13m_sel	**Select clk_26m_13m_ck** 0: clk26m 1: clk26m_d4
17:16	clk_ms_card_sel	**Selects f_fms_card_ck clock mux** 0: clk26m_d2 1: univpll3_d8 2: syspll4_d4 3: clk_26m_13m_ck
15	pdn_host_spi	**Turns off hf_fhost_spi_ck** 1: Enable clock-off
12	clk_host_spi_inv	**Inverts hf_fhost_spi_ck clock phase** 1: Enable phase inversion
10:8	clk_host_spi_sel	**Selects hf_fintdir_ck clock mux** 0: clk26m 1: syspll3_d4 2: syspll2_d4 3: syspll4_d2 4: syspll1_d4 5: univpll2_d2 6: univpll3_d2 7: syspll2_d8
7	pdn_asm_h	**Turns off f_fasm_h_ck** 1: Enable clock-off
4	clk_asm_h_inv	**Inverts f_fasm_h_ck clock phase** 1: Enable phase inversion
1:0	clk_asm_h_sel	**Selects f_fasm_h_ck clock mux** 0: clk26m 1: univpll2_d4 2: univpll2_d2 3: syspll_d5

100000D4 **CLK_CFG_13_SET** **SET control of CLK_CFG_13** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_13_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_13_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_13_set	**Sets the correspondent bit of CLG_CFG_SEL_13** 0: Unchanged 1: Set 1'b1 to the correspondent bit

100000D8 **CLK_CFG_13_CLR** **CLR control of CLK_CFG_13** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_13_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_13_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_13_clr	**Clears the correspondent bit of CLG_CFG_SEL_13** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000E0 **CLK_CFG_14** **Function clock selection register 14** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	pdn_spi2			clk_spi2_inv		clk_spi2_sel			pdn_cmsys			clk_cmsys_inv	clk_cmsys_sel			
Type	RW			RW		RW			RW			RW	RW			
Reset	0			0		0	0	0	0			0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_msdc30_3			clk_msdc30_3_inv		clk_msdc30_3_sel										clk_hdmirx_26m_24m_sel
Type	RW			RW		RW										RW
Reset	0			0		0	0	0								0

Bit(s)	Name	Description
31	pdn_spi2	**Turns off hf_fspi2_ck** 1: Enable clock-off
28	clk_spi2_inv	**Inverts hf_fspi2_ck clock phase** 1: Enable phase inversion
26:24	clk_spi2_sel	**Selects hf_fspi2_ck clock mux** 0: clk26m 1: syspll3_d2 2: syspll4_d2 3: univpll2_d4 4: univpll1_d8 5: clk26m 6: clk26m 7: clk26m
23	pdn_cmsys	**Turns off f_cmsys_ck** 1: Enable clock-off
20	clk_cmsys_inv	**Inverts hf_fcmsys_ck clock phase** 1: Enable phase inversion
19:16	clk_cmsys_sel	**Selects f_fcmsys_ck clock mux** 0: clk26m

Bit(s)	Name	Description
		1: syspll1_d2 2: univpll1_d2 3: univpll_d5 4: syspll_d5 5: syspll2_d2 6: syspll1_d4 7: syspll3_d2 8: syspll2_d4 9: syspll1_d8 10: syspll2_d8 11: clk26m 12: clk26m 13: clk26m 14: clk26m 15: clk26m
15	pdn_msdc30_3	**Turns off hf_fmsdc30_3_ck** 1: Enable clock-off
12	clk_msdc30_3_inv	**Inverts hf_fmsdc30_3_ck clock phase** 1: Enable phase inversion
10:8	clk_msdc30_3_sel	**Selects hf_fmsdc30_3_ck clock mux** 0: clk26m 1: msdcpll_ck 2: syspll2_d2 3: syspll1_d4 4: univpll1_d4 5: msdcpll_d2 6: msdcpll_d4 7: msdcpll_d8
0	clk_hdmirx_26m_24m_sel	**Selects f_fasm_h_ck clock mux** 0: clk26m 1: univpll_d52

100000E4 CLK_CFG_14_SET SET control of CLK_CFG_14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_14_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_14_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_14_set	**Sets the correspondent bit of CLG_CFG_SEL_14** 0: Unchanged 1: Set 1'b1 to the correspondent bit

100000E8 CLK_CFG_14_CLR CLR control of CLK_CFG_14 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	clk_cfg_14_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_14_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_14_clr	**Clears the correspondent bit of CLG_CFG_SEL_14** 0: Unchanged 1: Set 1'b0 to the correspondent bit

100000F0 CLK_CFG_15 Function clock selection register 15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									pdn_aud2dvd							clk_aud2dvd_sel
Type									RW							RW
Reset									0							0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	pdn_8bdac			clk_8bdac_inv			clk_8bdac_sel		pdn_spi1			clk_spi1_inv		clk_spi1_sel		
Type	RW			RW			RW		RW			RW		RW		
Reset	0			0			0	0	0			0		0	0	0

Bit(s)	Name	Description
23	pdn_aud2dvd	**Turns off f_faud2dvd_ck** 1: Enable clock-off
16	clk_aud2dvd_sel	**Selects f_faud2dvd_ck clock mux** 0: hf_fa1sys_hp_ck_d4 1: hf_fa2sys_hp_ck_d4
15	pdn_8bdac	**Turns off hf_f8bdac_ck** 1: Enable clock-off
12	clk_8bdac_inv	**Inverts hf_f8bdac_ck clock phase** 1: Enable phase inversion
9:8	clk_8bdac_sel	**Selects hf_f8bdac_ck clock mux** 0: clkrtc_int 1: f_8bdac_ck_pre 2: clk26m 3: clk26m
7	pdn_spi1	**Turns off f_fspi1_ck** 1: Enable clock-off
4	clk_spi1_inv	**Inverts f_fspi1_ck clock phase** 1: Enable phase inversion
2:0	clk_spi1_sel	**Selects f_fspi1_ck clock mux** 0: clk26m 1: syspll3_d2 2: syspll4_d2 3: univpll2_d4 4: univpll1_d8 5: clk26m

Bit(s)	Name	Description
		6: clk26m 7: clk26m

100000F4 CLK_CFG_15_SET SET control of CLK_CFG_15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_15_set[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_15_set[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_15_set	**Sets the correspondent bit of CLG_CFG_SEL_14** 0: Unchanged 1: Set 1'b1 to the correspondent bit

100000F8 CLK_CFG_15_CLR CLR control of CLK_CFG_15 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	clk_cfg_15_clr[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	clk_cfg_15_clr[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	clk_cfg_15_clr	**Clears the correspondent bit of CLG_CFG_SEL_14** 0: Unchanged 1: Set 1'b0 to the correspondent bit

10000100 CLK_CFG_8 Function clock selection register 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			abist_clk_sel									clk_pad_mclk_inv		clk_padmclk_sel		
Type			RW									RW				
Reset			0	0	0	0	0	0				0		0	0	0

Bit(s)	Name	Description
13:8	abist_clk_sel	**Selects f_fabist_ck clock mux (not glitch free)** 01: AD_MAIN_H546M_CK 02: AD_MAIN_H364M_CK 03: AD_MAIN_H218P4M_CK 04: AD_MAIN_H156M_CK 05: AD_UNIV_624M_CK 06: AD_UNIV_416M_CK 07: AD_UNIV_249P6M_CK 08: AD_UNIV_178P3M_CK 09: AD_UNIV_48M_CK 10: AD_USB_48M_CK 11: AD_MMPLL_CK 12: AD_MSDCPLL_CK 13: AD_DPICLK 14: clkph_MCK_o 15: AD_MEMPLL2_CKOUT0_PRE_ISO 16: AD_HADDS2PLL_294M_CK 17: AD_AUD2PLL_270M_CK 18: AD_HDMI_RX_CK 19: AD_USB20_CLK480M 20: rtc32k_ck_i 21: AD_SYS_26M_CK 22: AD_VENCPLL_CK 23~32: Reserved 33: abist_clk1 (AD_MIPI_26M_CK) 34: abist_clk2 (AD_AUD1PLL_294M_CK) 35: abist_clk3 (AD_MEM_26M_CK) 36: abist_clk4 (AD_PLLGP_TST_CK) 37: abist_clk5 (AD_DSI0_LNTC_DSICLK) 38: abist_clk6 (AD_MPPLL_TST_CK) 39: abist_clk7 (armpll_occ_mon) 40: abist_clk8 (AD_MEM2MIPI_26M_CK) 41: abist_clk9 (AD_MEMPLL_MONCLK) 42: abist_clk10 (AD_MEMPLL2_MONCLK) 43: abist_clk11 (AD_MEMPLL3_MONCLK) 44: abist_clk12 (AD_MEMPLL4_MONCLK) 45: abist_clk13 (AD_MEMPLL2_REFCLK) 46: abist_clk14 (AD_MEMPLL2_FBCLK) 47: abist_clk15 (AD_BBDAC_BGCHOP_26M_CLK) 48: abist_clk16 (AD_AUD1PLL_98M_CK) 49: abist_clk17 (AD_AUD2PLL_90M_CK) 50: abist_clk18 (AD_ETHPLL_500M_CK) 51: abist_clk19 (AD_HADDS2_98M_CLK) 52: abist_clk20 (AD_HDMI_0_ABIST_300MCK) 53: abist_clk21 (AD_HDMIRX_26M_CLK) 54: abist_clk22 (AD_TVDPLL_CK) 55: abist_clk23 (AD_TVD2PLL_CK) 56: abist_clk24 (AD_HDMITX_MONCLK) 57: abist_clk25 (Reserved) 58: abist_clk26 (AD_TRMII_CK) 59: abist_clk27 (Reserved) 60: abist_clk28 (Reserved) 61: abist_clk29 (AD_TVDPLL_CK) 62: abist_clk30 (AD_AUD1PLL_98M_CK) 63: abist_clk31 (AD_LVDSPLL_ETH_CK)
4	clk_padmclk_inv	**Inverts hf_padmclk_ck clock phase** 1: Enable phase inversion
2:0	clk_padmclk_sel	**select clk_padmclk clock mux** 0: clk26m 1: univpll_d26

Bit(s)	Name	Description
		2: univpll_d52 3: univpll_d108 4: univpll2_d8 5: univpll2_d16 6: univpll2_d32

10000104 CLK_CFG_9 Function clock selection register 9 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											ckgen_clk_sel					
Type											RW					
Reset											0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
21:16	ckgen_clk_sel	**Selects f_fckgen_ck clock mux (not glitch free)** 01: hf_faxi_ck 02: hd_faxi_ck 03: hf_fnfi2x_ck 04: hf_fddrphycfg_ck 05: hf_fmm_ck 06: f_fpwm_ck 07: hf_fvdec_ck 08: hf_fmfg_ck 09: hf_fcamtg_ck 10: f_fuart_ck 11: hf_fspi0_ck 12: f_fusb20_ck 13: hf_fmsdc30_0_ck 14: hf_fmsdc30_1_ck 15: hf_fmsdc30_2_ck 16: hf_faudio_ck 17: hf_faud_intbus_ck 18: hf_fpmicspi_ck 19: f_frtc_ck 20: f_f26m_ck 21: f_f32k_md1_ck 22: f_frtc_conn_ck 23: hf_feth_50m_ck 24: hf_emmc_hclk_ck 25: hd_haxi_nli_ck 26: hd_qaxidcm_ck 27: f_ffpc_ck 28: hf_fdpi0_ck 29: f_fckbus_ck_scan 30: f_fckrtc_ck_scan 31: hf_fdpilvds_ck 32: hf_fflash_ck 33: hf_fdi_ck 34: hf_fnr_ck 35: hf_fosd_ck 36: hf_fhdmirx_bist_ck 37: hf_fa1sys_hp_ck 38: hf_fa2sys_hp_ck 39: hf_fi2s1_mclk

Bit(s)	Name	Description
		40: hf_fi2s2_mclk 41: hf_fi2s3_mckj 42: hf_fi2s4_mclk 43: hf_fi2s5_mclk 44: hf_fi2s6_mclk 45: hf_fintdir_ck 46: hf_fasm_l_ck 47: hf_fasm_m_ck 48: hf_fasm_h_ck 49: hf_fhost_spi_ck 50: hf_fspi1_ck 51: hf_fspi2_ck 52: hf_fmsdc30_3_ck 53: hf_f8bdac_ck

10000108 CLK_CFG_10 Debug monitor clock selection register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													clk_ckmon3_sel			
Type													RW			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					clk_ckmon2_sel								clk_ckmon1_sel			
Type					RW								RW			
Reset					0	0	0	0					0	0	0	0

Bit(s)	Name	Description
19:16	clk_ckmon3_sel	**Selects f_fckmon3_ck clock mux (not glitch free)** 1: AD_SYS_26M_CK 2: rtc32k_ck_i 3: AD_WHPLL_2505P25M_CK (abist_clk_en = 1'b1) 4: AD_WPLL_245P76M_CK (abist_clk_en = 1'b1) 5: AD_MDPLL1_416M_CK (abist_clk_en = 1'b1) 6: AD_MCUPLL1_H481M_CK (abist_clk_en = 1'b1) 7: clkph_MCLK_o 8: AD_DPICLK 9: AD_MSDCPLL_CK 10: AD_MMPLL_CK 11: AD_UNIV_178P3M_CK 12: AD_MAIN_H156M_CK 13: AD_VENCPLL_CK 14: AD_LVDSPLL_CK 15: AD_LVDSPLL_ETH_CK
11:8	clk_ckmon2_sel	**Selects f_fckmon2_ck clock mux (not glitch free)** 1: AD_SYS_26M_CK 2: rtc32k_ck_i 3: AD_WHPLL_2505P25M_CK (abist_clk_en = 1'b1) 4: AD_WPLL_245P76M_CK (abist_clk_en = 1'b1) 5: AD_MDPLL1_416M_CK (abist_clk_en = 1'b1) 6: AD_MCUPLL1_H481M_CK (abist_clk_en = 1'b1) 7: clkph_MCLK_o 8: AD_DPICLK 9: AD_MSDCPLL_CK 10: AD_MMPLL_CK 11: AD_UNIV_178P3M_CK 12: AD_MAIN_H156M_CK 13: AD_VENCPLL_CK 14: AD_LVDSPLL_CK

Bit(s)	Name	Description
		15: AD_LVDSPLL_ETH_CK
3:0	clk_ckmon1_sel	**Selects f_fckmon1_ck clock mux (not glitch free)** 1: AD_SYS_26M_CK 2: rtc32k_ck_i 3: AD_WHPLL_2505P25M_CK (abist_clk_en = 1'b1) 4: AD_WPLL_245P76M_CK (abist_clk_en = 1'b1) 5: AD_MDPLL1_416M_CK (abist_clk_en = 1'b1) 6: AD_MCUPLL1_H481M_CK (abist_clk_en = 1'b1) 7: clkph_MCLK_o 8: AD_DPICLK 9: AD_MSDCPLL_CK 10: AD_MMPLL_CK 11: AD_UNIV_178P3M_CK 12: AD_MAIN_H156M_CK 13: AD_VENCPLL_CK 14: AD_LVDSPLL_CK 15: AD_LVDSPLL_ETH_CK

1000010C CLK_CFG_11 Debug monitor divider control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									ckmon3_k1							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ckmon2_k1								ckmon1_k1							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23:16	ckmon3_k1	
15:8	ckmon2_k1	
7:0	ckmon1_k1	

10000120 CLK_AUDDIV_0 audio clock divider control register0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	audio_a2sys_hp_k1								audio_a1sys_hp_k1							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	audio_ext2_k1								audio_ext1_k1							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	audio_a2sys_hp_k1	**Divider setting of a2sys_hp_ck**
23:16	audio_a1sys_hp_k1	**divider setting of a1sys_hp_ck**
15:8	audio_ext2_k1	**divider setting of audio_ext2_div_ck**
7:0	audio_ext1_k1	**divider setting of audio_ext1_div_ck**

10000124 CLK_AUDDIV_1 audio clock divider control register1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	audio_k4								audio_k3							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	audio_k2								audio_k1							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	audio_k4	**divider setting of i2s4_mclk**
23:16	audio_k3	**divider setting of i2s3_mclk**
15:8	audio_k2	**divider setting of i2s2_mclk**
7:0	audio_k1	**divider setting of i2s1_mclk**

10000128 CLK_AUDDIV_2 audio clock divider control register2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	audio_k6								audio_k5							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:8	audio_k6	**divider setting of i2s6_mclk**
7:0	audio_k5	**Divider setting of i2s5_mclk**

1000012C CLK_AUDDIV_3 audio clock divider control register3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		pdn_aud2_apll_ref	pdn_aud1_apll_ref	pdn_i2s6_mclk	pdn_i2s5_mclk	pdn_i2s4_mclk	pdn_i2s3_mclk	pdn_i2s2_mclk	pdn_i2s1_mclk	pdn_a2sys_hp	pdn_a1sys_hp	audio_k6_src_sel	audio_k5_src_sel	audio_k4_src_sel	audio_k3_src_sel	audio_k2_src_sel
Type		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	audio_k1_src_sel	aud2_apll_ref_mux_sel			aud1_apll_ref_mux_sel			audio_apll_mux_sel			audio_ck_mux2_sel			audio_ck_mux1_sel		
Type	RW	RW			RW			RW			RW			RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
30	pdn_aud2_apll_ref	**Enable clock off** 1: Enable aud2_apll_ref_ck off

Bit(s)	Name	Description
29	pdn_aud1_apll_ref	**Enable clock off** 1: Enable aud1_apll_ref_ck off
28	pdn_i2s6_mclk	**Enable clock off** 1: Enable i2s6_mclk off
27	pdn_i2s5_mclk	**Enable clock off** 1: Enable hf_fi2s5_mclk off
26	pdn_i2s4_m	**Enable clock off** 1: Enable hf_fi2s4_mclk off
25	pdn_i2s3_m	**Enable clock off** 1: Enable hf_fi2s3_mclk off
24	pdn_i2s2_m	**Enable clock off** 1: Enable hf_fi2s2_mclk off
23	pdn_i2s1_m	**Enable clock off** 1: Enable hf_fi2s1_mclk off
22	pdn_a2sys_hp	**Enable clock off** 1: Enable hf_fa2sys_hp_ck off
21	pdn_a1sys_hp	**Enable clock off** 1: Enable hf_fa1sys_hp_ck off
20	audio_k6_src_sel	**i2s6_mclk selection** 0: audio_mux1_ck 1: audio_mux2_ck
19	audio_k5_src_sel	**i2s5_mclk selection** 0: audio_mux1_ck 1: audio_mux2_ck
18	audio_k4_src_sel	**i2s4_mclk selection** 0: audio_mux1_ck 1: audio_mux2_ck
17	audio_k3_src_sel	**i2s3_mclk_selection** 0: audio_mux1_ck 1: audio_mux2_ck
16	audio_k2_src_sel	**i2s2_mclk selection** 0: audio_mux1_ck 1: audio_mux2_ck
15	audio_k1_src_sel	**i2s1_mclk selction** 0: audio_mux1_ck 1: audio_mux2_ck
14:12	aud2_apll_ref_mux_sel	**aud2_apll_ref_ck selection** 0: clk26m 1: ext_i2s1_mck 2: ext_i2s2_mck 3: ext_i2s3_mck 4: ext_i2s4_mck 5: ext_i2s5_mck 6: ext_i2s6_mck
11:9	aud1_apll_ref_mux_sel	**aud1_apll_ref_ck selection** 0: clk26m 1: ext_i2s1_mck 2: ext_i2s2_mck 3: ext_i2s3_mck 4: ext_i2s4_mck 5: ext_i2s5_mck 6: ext_i2s6_mck

Bit(s)	Name	Description
8:6	audio_apll_mux_sel	**audio_apll_mux_ck selection** 0: clk26m 1: AD_AUD1PLL_98M_CK 2: AD_AUD2PLL_90M_CK 3: AD_HADDS2PLL_98M_CK 4: audio_ext1_div_ck 5: audio_ext2_div_ck
5:3	audio_ck_mux2_sel	**audio_mux2_ck selection** 0: clk26m 1: AD_AUD1PLL_98M_CK 2: AD_AUD2PLL_90M_CK 3: AD_HADDS2PLL_98M_CK 4: audio_ext1_div_ck 5: audio_ext2_div_ck
2:0	audio_ck_mux1_sel	**audio_mux1_ck selection** 0: clk26m 1: AD_AUD1PLL_98M_CK 2: AD_AUD2PLL_90M_CK 3: AD_HADDS2PLL_98M_CK 4: audio_ext1_div_ck 5: audio_ext2_div_ck

10000150 **CLK_8BDAC_CFG** **8bdac clock divider control register** **00000033**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**Clk_8bdac_cfg**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1

Bit(s)	Name	Description
15:0	Clk_8bdac_cfg	**Divider setting of f_8bdac_ck**

10000200 **CLK_SCP_CFG_0** **SCP control register 0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							**sc_mac_26m_off_en**		**sc_conn_32k_off_en**		**sc_md_32k_off_en**	**sc_armck_off_en**		**sc_axick_off_en**	**sc_memck_off_en**	**sc_26ck_off_en**
Type							RW		RW		RW	RW		RW	RW	RW
Reset							0		0		0	0		0	0	0

Bit(s)	Name	Description
9	sc_mac_26m_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
7	sc_conn_32k_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
5	sc_md_32k_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
4	sc_armck_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
2	sc_axick_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
1	sc_memck_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
0	sc_26ck_off_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control

10000204 CLK_SCP_CFG_1 SCP control register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												sc_axick_dcm_dis_en				sc_axick_26m_sel_en
Type												RW				RW
Reset												0				0

Bit(s)	Name	Description
4	sc_axick_dcm_dis_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control
0	sc_axick_26m_sel_en	0: Disable scpsys clock-off control 1: Enable scpsys clock-off control

10000210 CLK_MISC_CFG_0 Internal clk_rtc divider control register 5D3F0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		clkrtc_int_en	clkrtc_int_residual													
Type		RW	RW													
Reset		1	0	1	1	1	0	1	0	0	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
30	clkrtc_int_en	**Enable internal clkrtc**
29:16	clkrtc_int_residual	Setting residual of internal clkrtc divider Ex: 26M/32768 = 793.45703125

10000214 CLK_MISC_CFG_1 Frequency meter divider control register FF00FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ckgen_k1															
Type	RW															
Reset	1	1	1	1	1	1	1	1								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	arm_k1								abist_k1							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	ckgen_k1	**Divider setting of f_fckgen_ck**
15:8	arm_k1	**Divider setting of hf_farm_ck**
7:0	abist_k1	**Divider setting of f_fabist_ck**

10000220 CLK26CALI_0 Frequency meter control register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									fmeter_en		ckgen_clk_exc	ckgen_tri_cal		abist_clk_exc	pll_test	abist_tri_cal
Type									RW		RW	RW		RW	RW	RW
Reset									0		0	0		0	0	0

Bit(s)	Name	Description
7	fmeter_en	0: Disable fmeter 1: Enable fmeter
5	ckgen_clk_exc	**Selects measuring clock** 0: f_fckgen_ck 1: CLK26M
4	ckgen_tri_cal	**Triggers frequency meter on f_fckgen_ck.** Auto-cleared when calibration is done. 0: Disable 1: Enable
2	abist_clk_exc	**Selects measuring clock** 0: f_fabist_ck 1: CLK26M
1	pll_test	**Selects clock divider test clock** 0: PLL 1: Test clock (26MHz)

Bit(s)	Name	Description
0	abist_tri_cal	**Triggers frequency meter on f_fabist_ck** Auto-cleaedr when calibration is done. 0: Disable 1: Enable

10000224 CLK26CALI_1 Frequency meter control register 1 03FF0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							abist_load_cnt									
Type							RW									
Reset							1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	cal_cnt															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:16	abist_load_cnt	**Set to 0x3FF if ABIST is used.**
15:0	cal_cnt	**Frequency meter result of f_fabist_ck** Frequency = (26MHz*cal_cnt)/1024

10000228 CLK26CALI_2 Frequency meter control register 2 03FF0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							ckgen_load_cnt									
Type							RW									
Reset							1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ckgen_cal_cnt															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
25:16	ckgen_load_cnt	**Set to 0x3FF if fmeter is used**
15:0	ckgen_cal_cnt	**Frequency meter result of f_fckgen_ck** Frequency = (26MHz*cal_cnt)/1024

1000022C CKSTA_REG Function clock selection status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																chg_sta
Type																RU
Reset																0

Bit(s)	Name	Description

Bit(s)	Name	Description
0	chg_sta	Clock switches changing in progress

10000230 **TEST_MODE_CFG** **Test mode control register** **00000200**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					rg_conn_top2_test_mem_pd	rg_conn_top1_test_mem_pd	rg_conn_top2_test_pwr_on									
Type					RW	RW	RW									
Reset					0	0	1									

Bit(s)	Name	Description
11	rg_conn_top2_test_mem_pd	
10	rg_conn_top1_test_mem_pd	
9	rg_conn_top2_test_pwr_on	

1000030C **MBIST_CFG_1** **Debug monitor selection register 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										ckgen_byte_sel						
Type										RW						
Reset										0	0	0				

Bit(s)	Name	Description
6:4	ckgen_byte_sel	Selects TOPCKGEN debugging monitor

10000310 **RESET_DEGLITCH_KEY** **Reset deglitch enable key register** **FFFFFFFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dgrst_en_key[31:16]															
Type	RW															
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dgrst_en_key[15:0]															
Type	RW															

Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit(s)	Name	Description
31:0	dgrst_en_key	**Write 0x67D2_A357 to enable reset deglitch.**

10000314 MBIST_CFG_3 Debug monitor selection register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	usb_gpio_reg															
Type	RW															
Reset	0	0	0	0	0	0	0	0								

Bit(s)	Name	Description
15:8	usb_gpio_reg	**Selects usb_i2c_mode**

10000318 BOOT_TRAP Boot strap register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															Boot_strap	
Type															RO	
Reset															0	0

Bit(s)	Name	Description
1:0	Boot_strap	**Strapping boot**

2.9 Programming Guide

2.9.1 Clock Off

- The clock can be turned on/off through changing the value of pdn_*. However, this control cannot be switched along with clk_*_sel and clk_*_inv.
- It is recommended to change pdn_* with SET and CLEAR function provided by CLK_CFG_*_SET and CLK_CFG_*_CLR. Because there may be clock with multi-bit pdn_* which are planned to avoid read modify write from different sub-systems (APSYS, MDSYS and CONNSYS) at the same time.
- SET and CLEAR function of CLK_CFG_* is a solution to avoid read modify write from different sub-systems.

2.9.2 Clock Switching

- Make sure clock A and clock B are available before changing the setting of clk_*_sel. If switched to a non-exist clock, the clock switch will be stuck until non-exist clock is turned on to free the clock switch.
- Supports multi-clock switching at the same time (without changing pdn_*)

Switching from clock A to clock B

1. Make sure clock B is ready.
2. Change clk_*_sel.
3. Wait until chg_sta = 1'b0 (optional).
4. Turn off clock A (optional).

2.9.3 Switch AXI to 26MHz by SCPSYS

The reflection time is about 17T 26MHz if all clocks are counted as 26MHz. Refer to the following formula:

*4T Bus Clock(*1) + 4T Current Clock(*2) + 5T Reference Clock(*3) + 1T Bus Clock(*4) + 3T Target Clock(*5)*

Comment	Description
Bus clock	26MHz (in current project)
Ref clock	26MHz, balance with bus clock
*1	2T Sync + 1 T Control
*2	3T sync
*3	4T sync
*4	1T control. For async CLKSW like hf_fmem_ck used, it will be 2T sync.
*5	2T sync

2.9.4 Frequency Meter

There are two frequency meters embedded inside TOPCKGEN.

1. Set fmeter_en to 1'b1.
2. Choose target clock by changing abist_clk_sel / ckgen_clk_sel.
3. Change abist_k1 / ckgen_k1 for dividing target clock (optional).
4. Change reference clock by changing clk_exec / ckgen_exec (optional).
5. Trigger frequency meter by set tri_cal / ckgen_tri_cal = 1b'1.
6. Wait until tri_cal / ckgen_tri_cal = 1'b0.
7. Read frequency meter result from cal_cnt / ckgen_cal_cnt.

- freq(target) = (abist_k1 + 1)*[freq(reference clock)*cal_cnt]/1024
- freq(target) = (ckgen_k1 + 1)*[freq(reference clock)*ckgen_cal_cnt]/1024

2.10 PLL Related Register Definition

Module name: APMIXEDSYS Base address: (+10209000h)

Address	Name	Width	Register Function
10209000	AP_PLL_CON0	32	AP PLL Control Register 0
10209004	AP_PLL_CON1	32	AP PLL Control Register 1
10209008	AP_PLL_CON2	32	AP PLL Control Register 2
10209014	PLL_HP_CON0	32	PLL Hopping Control Register 0
10209038	PLL_TEST_CON0	32	PLL Test Control Register 0
10209100	HDMI_CON0	32	HDMI control register 0
10209104	HDMI_CON1	32	HDMI control register 1
10209108	HDMI_CON2	32	HDMI control register 2
1020910C	HDMI_CON3	32	HDMI control register 3
10209110	HDMI_CON4	32	HDMI control register 4
10209114	HDMI_CON5	32	HDMI control register 5
10209118	HDMI_CON6	32	HDMI control register 6
1020911C	HDMI_CON7	32	HDMI control register 7
10209120	HDMI_CON8	32	HDMI control register 8
10209200	ARMPLL_CON0	32	ARMPLL Control Register 0
10209204	ARMPLL_CON1	32	ARMPLL Control Register 1
1020920C	ARMPLL_PWR_CON0	32	ARMPLL Power Control Register 0
10209210	MAINPLL_CON0	32	MAINPLL Control Register 0
10209214	MAINPLL_CON1	32	MAINPLL Control Register 1
1020921C	MAINPLL_PWR_CON0	32	MAINPLL Power Control Register 0
10209220	UNIVPLL_CON0	32	UNIVPLL Control Register 0
10209224	UNIVPLL_CON1	32	UNIVPLL Control Register 1
1020922C	UNIVPLL_PWR_CON0	32	UNIVPLL Power Control Register 0
10209230	MMPLL_CON0	32	MMPLL Control Register 0
10209234	MMPLL_CON1	32	MMPLL Control Register 1
1020923C	MMPLL_PWR_CON0	32	MMPLL Power Control Register 0
10209240	MSDCPLL_CON0	32	MSDCPLL Control Register 0
10209244	MSDCPLL_CON1	32	MSDCPLL Control Register 1
1020924C	MSDCPLL_PWR_CON0	32	MSDCPLL Power Control Register 0
10209250	TVDPLL_CON0	32	TVDPLL Control Register 0
10209254	TVDPLL_CON1	32	TVDPLL Control Register 1
1020925C	TVDPLL_PWR_CON0	32	TVDPLL Power Control Register 0
10209260	TVDPLL_SSC_C	32	TVDPLL SSC Control Register 0

	ON0		
10209264	TVDPLL_SSC_CON1	32	TVDPLL SSC Control Register 1
10209270	AUD1PLL_CON0	32	AUD1PLL Control Register 0
10209274	AUD1PLL_CON1	32	AUD1PLL Control Register 1
1020927C	AUD1PLL_PWR_CON0	32	AUDPLL Power Control Register 0
10209280	TRGPLL_CON0	32	TRGPLL Control Register 0
10209284	TRGPLL_CON1	32	TRGPLL Control Register 1
1020928C	TRGPLL_PWR_CON0	32	TRGPLL Power Control Register 0
10209290	ETHPLL_CON0	32	ETHPLL Control Register 0
10209294	ETHPLL_CON1	32	ETHPLL Control Register 1
1020929C	ETHPLL_PWR_CON0	32	ETHPLL Power Control Register 0
102092A0	VDECPLL_CON0	32	VDECPLL Control Register 0
102092A4	VDECPLL_CON1	32	VDECPLL Control Register 1
102092AC	VDECPLL_PWR_CON0	32	VDECPLL Power Control Register 0
102092B0	HADDS2PLL_CON0	32	HADDS2PLL Control Register 0
102092B4	HADDS2PLL_CON1	32	HADDS2PLL Control Register 1
102092BC	HADDS2PLL_PWR_CON0	32	HADDS2PLL Power Control Register 0
102092C0	AUD2PLL_CON0	32	AUD2PLL Control Register 0
102092C4	AUD2PLL_CON1	32	AUD2PLL Control Register 1
102092CC	AUD2PLL_PWR_CON0	32	AUD2PLL Power Control Register 0
102092D0	TVD2PLL_CON0	32	TVD2PLL Control Register 0
102092D4	TVD2PLL_CON1	32	TVD2PLL Control Register 1
102092DC	TVD2PLL_PWR_CON0	32	TVD2PLL Power Control Register 0
102092F0	TVD2PLL_SSC_CON0	32	TVD2PLL SSC Control Register 0
102092F4	TVD2PLL_SSC_CON1	32	TVD2PLL SSC Control Register 1
10209400	AP_AUXADC_CON0	32	AUXADC Control Register 0
10209404	AP_AUXADC_CON1	32	AUXADC Control Register 1
1020940C	AP_AUXADC_CON2	32	AUXADC Control Register 2
10209600	TS_CON0	32	Thermal Sensor Control Register 0
10209604	TS_CON1	32	Thermal Sensor Control Register 1
1000F800	VENCPLL_CON0	32	VENCPLL Control Register 0
1000F804	VENCPLL_CON1	32	VENCPLL Control Register 1
1000F80C	VENCPLL_PWR_CON0	32	VENCPLL Power Control Register 0

10209000 AP_PLL_CON0 AP PLL Control Register 0 7E00E131

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne		AP_DUMMY5	AP_DUMMY4	AP_DUMMY3	AP_DUMMY2	AP_DUMMY1	AP_DUMMY0	TVDPLL_REF_SEL								
Type		RW	RW	RW	RW	RW	RW	RW								
Reset		1	1	1	1	1	1	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	AUD1PLL_98M_EN	AUD2PLL_90M_EN	HADDS2PLL_98M_EN	AUD1PLL_REF_SEL	AUD2PLL_REF_SEL	CLKSQ1_HYS_SEL			TVD2PLL_REF_SEL	RG_SSUSB_SR_DRIVER_EN	MIPI_26M_OUT_EN	MEM_26M_OUT_EN	CLKSQ1_MON_EN	RG_SSUSB_SR_GATE_ENB	CLKSQ1_LPF_EN	CLKSQ1_EN
Type	RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	0	0	0	0	1	0	0	1	1	0	0	0	1

Bit(s)	Mnemonic	Name	Description
30	**AP_DUMMY5**	AP_DUMMY5	set this dummy bit to 0.
29	**AP_DUMMY4**	AP_DUMMY4	set this dummy bit to 0.
28	**AP_DUMMY3**	AP_DUMMY3	set this dummy bit to 0.
27	**AP_DUMMY2**	AP_DUMMY2	set this dummy bit to 0.
26	**AP_DUMMY1**	AP_DUMMY1	set this dummy bit to 0.
25	**AP_DUMMY0**	AP_DUMMY0	set this dummy bit to 0.
24	**TVDPLL_REF_SEL**	TVDPLL_REF_SEL	
15	**AUD1PLL_98M_EN**	AUD1PLL_98M_EN	
14	**AUD2PLL_90M_EN**	AUD2PLL_90M_EN	
13	**HADDS2PLL_98M_EN**	HADDS2PLL_98M_EN	
12	**AUD1PLL_REF_SEL**	AUD1PLL_REF_SEL	
11	**AUD2PLL_REF_SEL**	AUD2PLL_REF_SEL	
10:8	**CLKSQ1_HYS_SEL**	CLKSQ1_HYS_SEL	**Clock square hysterisis level selection**
7	**TVD2PLL_REF_SEL**	TVD2PLL_REF_SEL	
6	**RG_SSUSB_SR_DRIVER_EN**	RG_SSUSB_SR_DRIVER_EN	
5	**MIPI_26M_OUT_EN**	MIPI_26M_OUT_EN	**Enables MIPI 26MHz clock source**
4	**MEM_26M_OUT_EN**	MEM_26M_OUT_EN	**Enables DDR 26MHz clock source**
3	**CLKSQ1_MON_EN**	CLKSQ1_MON_EN	**Enables clock square monitor** 0: Disable 1: Enable

Bit(s)	Mnemonic	Name	Description
2	**RG_SSUSB_SR_GATE_ENB**	RG_SSUSB_SR_GATE_ENB	
1	**CLKSQ1_LPF_EN**	CLKSQ1_LPF_EN	**Enables clock square1 low-pass filter**
0	**CLKSQ1_EN**	CLKSQ1_EN	**Enables clock square1**

10209004 AP_PLL_CON1 AP PLL Control Register 1 00FF03F3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									MMPLL_ISO_SEL	UNIVPLL_ISO_SEL	MAINPLL_ISO_SEL	ARMPLL_ISO_SEL	MMPLL_PWR_SEL	UNIVPLL_PWR_SEL	MAINPLL_PWR_SEL	ARMPLL_PWR_SEL
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				ABI_REF_CLK_DIV_SLE			MAINDIV_RSTB_SEL	UNIVDIV_RSTB_SEL	MMPLL_EN_SEL	UNIVPLL_EN_SEL	MAINPLL_EN_SEL	ARMPLL_EN_SEL			CLKSQ1_LPF_EN_SEL	CLKSQ1_EN_SEL
Type				RW			RW	RW	RW	RW	RW	RW			RW	RW
Reset				0	0	0	1	1	1	1	1	1			1	1

Bit(s)	Mnemonic	Name	Description
23	**MMPLL_ISO_SEL**	MMPLL_ISO_SEL	**MMPLL ISO_EN control selection** 0: Sleep control 1: Register control
22	**UNIVPLL_ISO_SEL**	UNIVPLL_ISO_SEL	**UNIVPLL ISO_EN control selection** 0: Sleep control 1: Register control
21	**MAINPLL_ISO_SEL**	MAINPLL_ISO_SEL	**MAINPLL ISO_EN control selection** 0: Sleep control 1: Register control
20	**ARMPLL_ISO_SEL**	ARMPLL_ISO_SEL	**ARMPLL ISO_EN control selection** 0: Sleep control 1: Register control
19	**MMPLL_PWR_SEL**	MMPLL_PWR_SEL	**MMPLL PWR_ON control selection** 0: Sleep control 1: Register control
18	**UNIVPLL_PWR_SEL**	UNIVPLL_PWR_SEL	**UNIVPLL PWR_ON control selection** 0: Sleep control 1: Register control
17	**MAINPLL_PWR_SEL**	MAINPLL_PWR_SEL	**MAINPLL PWR_ON control selection** 0: Sleep control 1: Register control
16	**ARMPLL_PWR_SEL**	ARMPLL_PWR_SEL	**ARMPLL PWR_ON control selection** 0: Sleep control 1: Register control
12:10	**ABI_REF_CLK_DIV_SLE**	ABI_REF_CLK_DIV_SLE	
9	**MAINDIV_R**	MAINDIV_RSTB_S	**MAINPLL DIV_RSTB control selection**

Bit(s)	Mnemonic	Name	Description
	STB_SEL	EL	0: Sleep control 1: Register control
8	**UNIVDIV_RSTB_SEL**	UNIVDIV_RSTB_SEL	**UNIVPLL DIV_RSTB control selection** 0: Sleep control 1: Register control
7	**MMPLL_EN_SEL**	MMPLL_EN_SEL	**MMPLL enable control selection** 0: Sleep control 1: Register control
6	**UNIVPLL_EN_SEL**	UNIVPLL_EN_SEL	**UNIVPLL enable control selection** 0: Sleep control 1: Register control
5	**MAINPLL_EN_SEL**	MAINPLL_EN_SEL	**MAINPLL enable control selection** 0: Sleep control 1: Register control
4	**ARMPLL_EN_SEL**	ARMPLL_EN_SEL	**ARMPLL enable control selection** 0: Sleep control 1: Register control
1	**CLKSQ1_LPF_EN_SEL**	CLKSQ1_LPF_EN_SEL	**CLKSQ LPF_EN control selection** 0: Sleep control 1: Register control
0	**CLKSQ1_EN_SEL**	CLKSQ1_EN_SEL	**CLKSQ enable control selection** 0: Sleep control 1: Register control

10209008 AP_PLL_CON2 AP PLL Control Register 2 00000007

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										**UNIVPLL_OUT_OFF**	**MAINPLL_OUT_OFF**	**ARMPLL_OUT_OFF**		**UNIVPLL_OUT_OFF_SEL**	**MAINPLL_OUT_OFF_SEL**	**ARMPLL_OUT_OFF_SEL**
Type										RW	RW	RW		RW	RW	RW
Reset										0	0	0		1	1	1

Bit(s)	Mnemonic	Name	Description
6	**UNIVPLL_OUT_OFF**	UNIVPLL_OUT_OFF	**UNIVPLL CG register control**
5	**MAINPLL_OUT_OFF**	MAINPLL_OUT_OFF	**MAINPLL CG register control**
4	**ARMPLL_OUT_OFF**	ARMPLL_OUT_OFF	**ARMPLL CG register control**
2	**UNIVPLL_OUT_OFF_SEL**	UNIVPLL_OUT_OFF_SEL	**UNIVPLL CG control selection** 0: Sleep control 1: Register control
1	**MAINPLL_OUT_OFF_SEL**	MAINPLL_OUT_OFF_SEL	**MAINPLL CG control selection** 0: Sleep control 1: Register control

Bit(s)	Mnemonic	Name	Description
0	**ARMPLL_OUT_OFF_SEL**	ARMPLL_OUT_OFF_SEL	**ARMPLL CG control selection** 0: Sleep control 1: Register control

10209014 PLL_HP_CON0 PLL Hopping Control Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										TVDPLL_HP_EN	VENCPLL_HP_EN	MSDCPLL_HP_EN	MEMPLL_HP_EN	MMPLL_HP_EN	MAINPLL_HP_EN	ARMPLL_HP_EN
Type										RW	RW	RW	RW	RW	RW	RW
Reset										0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
6	**TVDPLL_HP_EN**	TVDPLL_HP_EN	**TVDPLL hopping control** 0: Register control 1: Hopping control
5	**VENCPLL_HP_EN**	VENCPLL_HP_EN	**VENCPLL hopping control** 0: Register control 1: Hopping control
4	**MSDCPLL_HP_EN**	MSDCPLL_HP_EN	**MSDCPLL hopping control** 0: Register control 1: Hopping control
3	**MEMPLL_HP_EN**	MEMPLL_HP_EN	**MEMPLL hopping control** 0: Register control 1: Hopping control
2	**MMPLL_HP_EN**	MMPLL_HP_EN	**MSDCPLL hopping control** 0: Register control 1: Hopping control
1	**MAINPLL_HP_EN**	MAINPLL_HP_EN	**MAINPLL hopping control** 0: Register control 1: Hopping control
0	**ARMPLL_HP_EN**	ARMPLL_HP_EN	**ARMPLL hopping control** 0: Register control 1: Hopping control

10209038 PLL_TEST_CON0 PLL Test Control Register 0 00C00000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									PLLGP_RESERVE							
Type									RW							
Reset									1	1	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			PLLDIV_TEST	PLLGP_MONREF_EN	PLLGP_LVROD_EN	PLLGP_TST_SEL			PLLGP_TSTMUX				PLLGP_TSTOD_EN	PLLGP_A2DCK_EN	PLLGP_TSTCK_EN	PLLGP_TST_EN

Type			RW	RW	RW	RW			RW				RW	RW	RW	RW
Reset			0	0	0	0	0		0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23:16	**PLLGP_RESERVE**	PLLGP_RESERVE	
13	**PLLDIV_TEST**	PLLDIV_TEST	
12	**PLLGP_MONREF_EN**	PLLGP_MONREF_EN	**FBK clock testing mode enable**
11	**PLLGP_LVROD_EN**	PLLGP_LVROD_EN	**Overdrive open drain LDO**
10:9	**PLLGP_TST_SEL**	PLLGP_TST_SEL	**Test Mux Selection**
7:4	**PLLGP_TSTMUX**	PLLGP_TSTMUX	**Test Mux Selection**
3	**PLLGP_TSTOD_EN**	PLLGP_TSTOD_EN	**Enable open drain**
2	**PLLGP_A2DCK_EN**	PLLGP_A2DCK_EN	**Enable frequency meter path**
1	**PLLGP_TSTCK_EN**	PLLGP_TSTCK_EN	**PLL output clock testing mode enable**
0	**PLLGP_TST_EN**	PLLGP_TST_EN	**Testing mode enable**

10209100 HDMI_CON0 HDMI control register 0 0000000A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HDMITX_EN_DRV				RG_HDMITX_EN_IMP				RG_HDMITX_EN_PRED				RG_HDMITX_EN_SLDO			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_HDMITX_EN_SER				RG_HDMITX_EN_DIN_BIST	RG_HDMITX_EN_SER_PEM	RG_HDMITX_EN_SER_ABIST	RG_HDMITX_EN_SER_ABEDG	RG_HDMITX_SER_PASS_SEL		RG_HDMITX_DRV_IBIAS					
Type	RW				RW	RW	RW	RW	RW		RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
31:28	**RG_HDMITX_EN_DRV**	RG_HDMITX_EN_DRV	**enable TX DRV** 1'b0: disable 1'b1: enable
27:24	**RG_HDMITX_EN_IMP**	RG_HDMITX_EN_IMP	**enable TX IMP** 1'b0: disable 1'b1: enable
23:20	**RG_HDMITX_EN_PRED**	RG_HDMITX_EN_PRED	**enable TX Predriver** 1'b0: disable 1'b1: enable
19:16	**RG_HDMITX_EN_SLDO**	RG_HDMITX_EN_SLDO	**enable TX SER2T1 LDO** 1'b0: disable 1'b1: enable

Bit(s)	Mnemonic	Name	Description
15:12	**RG_HDMITX_EN_SER**	RG_HDMITX_EN_SER	**enable TX SER** 1'b0: disable 1'b1: enable
11	**RG_HDMITX_EN_DIN_BIST**	RG_HDMITX_EN_DIN_BIST	**enable TX data bist gen to output random data** 1'b0: disable 1'b1: enable
10	**RG_HDMITX_EN_SER_PEM**	RG_HDMITX_EN_SER_PEM	**enable TX SER Pre-ephasis path** 1'b0: disable 1'b1: enable
9	**RG_HDMITX_EN_SER_ABIST**	RG_HDMITX_EN_SER_ABIST	**NC**
8	**RG_HDMITX_EN_SER_ABEDG**	RG_HDMITX_EN_SER_ABEDG	**enable TX SER Abist edge detect** 1'b0: disable 1'b1: enable
7:6	**RG_HDMITX_SER_PASS_SEL**	RG_HDMITX_SER_PASS_SEL	**select Predriver input from test_in** [1]=1'b0: Predirver input from SER [1]=1'b1: Predirver input from test_in [0]=1'b0: test_in from PLL HR CLK [0]=1'b1: test_in from mon_clk
5:0	**RG_HDMITX_DRV_IBIAS**	RG_HDMITX_DRV_IBIAS	**set TX DRV bias current** [5]: enable 0.5mA [4]: enable 5mA [3]: enable 8mA [2]: enable 4mA [1]: enable 2mA [0]: enable 1mA if IBIAS[5:0]=6'b111111, I=20.5mA if IBIAS[5:0]=6'b001010, I=10mA

10209104 HDMI_CON1 HDMI control register 1 002C0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HDMITX_DRV_IMP						RG_HDMITX_SER_BIST_TOG	RG_HDMITX_SER_CLKDIG_INV	RG_HDMITX_SER_DIN_SEL	RG_HDMITX_PRED_IMP	RG_HDMITX_PRED_IBIAS				RG_HDMITX_CKLDO_LVROD	
Type	RW						RW	RW	RW	RW	RW				RW	
Reset	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_HDMITX_SLDO_LVROD		RG_HDMITX_SER_DIN													
Type	RW		RW													
Reset	0	0	0	0	0	0	0	0	0	0	0	0				

Bit(s)	Mnemonic	Name	Description
31:26	**RG_HDMITX_DRV_IMP**	RG_HDMITX_DRV_IMP	**set TX DRV impedance=4.2k/ ([4:0]+20x[5])** [5]: enable 4.2k/20 [4]: enable 4.2k/16 [3]: enable 4.2k/8 [2]: enable 4.2k/4 [1]: enable 4.2k/2 [0]: enable 4.2k/1

Bit(s)	Mnemonic	Name	Description
			if IMP[5:0]=6'b011100, DRV impedance=4.2k/28=150
25	**RG_HDMITX_SER_SER_BIST_TOG**	RG_HDMITX_SER_BIST_TOG	**set data toggle for ABIST** 1'b0: disable 1'b1: enable
24	**RG_HDMITX_SER_CLKDIG_INV**	RG_HDMITX_SER_CLKDIG_INV	**set CLKDIG & CLKDIG_CTS same/invert phase to CK_ANAFF** 1'b0: same phase 1'b1: invert phase
23	**RG_HDMITX_SER_DIN_SEL**	RG_HDMITX_SER_DIN_SEL	**select SER data source** 1'b0: from digital 1'b1: from RG_SER_DIN[9:0]
22	**RG_HDMITX_PRED_IMP**	RG_HDMITX_PRED_IMP	**set Predriver impedance** 1'b0: 200 ohm for <1.5Gbps 1'b1: 100 ohm for >1.5Gbps ***note for swing (1)<1.5Gbps, set PRED_IMP=1'b0 PRED_IBAS[3:0]=4'b1011 (1)>1.5Gbps, set PRED_IMP=1'b1 PRED_IBAS[3:0]=4'b1111
21:18	**RG_HDMITX_PRED_IBIAS**	RG_HDMITX_PRED_IBIAS	**set Predirver bias current** [3]: enable 0.5mA [2]: enable 4mA [1]: enable 2mA [0]: enable 1mA if IBIAS[3:0]=4'b1011, current=3.5mA ***note for swing (1)<1.5Gbps, set PRED_IMP=1'b0 PRED_IBAS[3:0]=4'b1011 (1)>1.5Gbps, set PRED_IMP=1'b1 PRED_IBAS[3:0]=4'b1111
17:16	**RG_HDMITX_CKLDO_LVROD**	RG_HDMITX_CKLDO_LVROD	**Clock LDO voltage boost** 2'b00: LDO=VCCK 2'b01: LDO=VCCK+50mV 2'b10: LDO=VCCK+50mV 2'b11: LDO=VCCK+100mV
15:14	**RG_HDMITX_SLDO_LVROD**	RG_HDMITX_SLDO_LVROD	**SER2T1 LDO voltage boost** 2'b00: LDO=VCCK 2'b01: LDO=VCCK+50mV 2'b10: LDO=VCCK+50mV 2'b11: LDO=VCCK+100mV
13:4	**RG_HDMITX_SER_DIN**	RG_HDMITX_SER_DIN	**if RG_HDMITX_SER_DIN_SEL=1, SER 10 bit data comes from this register value**

10209108 HDMI_CON2 HDMI control register 2 0007C000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HDMITX_SER_TEST_SEL								RG_HDMITX_DATA_CLKCH[9:2]							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_HDMITX_DATA_CLK		RG_HDMITX_MBIAS_LOI		RG_HDMITX_MBIAS_LOX		RG_HDMITX	RG_HDMITX	RG_HDMITX	RG_HDMITX	RG_HDMITX	RG_HDMITX_TX_POSDIV		RG_HDMITX	RG_HDMITX	RG_HDMITX

	CH[1:0]		BI		BI		_EN_MBIAS_LOI	_EN_MBIAS_LOX	_MBIAS_LPF_EN	_EN_MBIAS	_TX_POSDIV_SEL			_REFEXT_SEL	_EN_TX_POSDIV	_EN_TX_CKLDO
Type	RW		RW		RW		RW	RW	RW	RW	RW	RW		RW	RW	RW
Reset	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**RG_HDMITX_SER_TEST_SEL**	RG_HDMITX_SER_TEST_SEL	
23:14	**RG_HDMITX_DATA_CLKCH**	RG_HDMITX_DATA_CLKCH	**PAD_CLK data from this register value**
13:12	**RG_HDMITX_MBIAS_LOIBI**	RG_HDMITX_MBIAS_LOIBI	**set INTR local bias current** 2'b00: 80uA 2'b01: 100uA 2'b10: 100uA 2'b11: 133uA
11:10	**RG_HDMITX_MBIAS_LOXBI**	RG_HDMITX_MBIAS_LOXBI	**set EXTR local bias current** 2'b00: 80uA 2'b01: 100uA 2'b10: 100uA 2'b11: 133uA
9	**RG_HDMITX_EN_MBIAS_LOI**	RG_HDMITX_EN_MBIAS_LOI	**enable INTR current from local bias** 1'b0: disable 1'b1: enable
8	**RG_HDMITX_EN_MBIAS_LOX**	RG_HDMITX_EN_MBIAS_LOX	**enable EXTR current from local bias** 1'b0: disable 1'b1: enable
7	**RG_HDMITX_MBIAS_LPF_EN**	RG_HDMITX_MBIAS_LPF_EN	**enable Mbias low-pass filter** 1'b0: disable 1'b1: enable
6	**RG_HDMITX_EN_MBIAS**	RG_HDMITX_EN_MBIAS	**enable Mbias current** 1'b0: disable 1'b1: enable
5	**RG_HDMITX_TX_POSDIV_SEL**	RG_HDMITX_TX_POSDIV_SEL	**sel TX half-rate clock source** 1'b0: from PLL/1 1'b1: from PLL/POSDDIV[1:0]
4:3	**RG_HDMITX_TX_POSDIV**	RG_HDMITX_TX_POSDIV	**TX half-rate post divider setting** 2'b00: /1 2'b01: /2 2'b10: /4 2'b11: NC
2	**RG_HDMITX_REFEXT_SEL**	RG_HDMITX_REFEXT_SEL	
1	**RG_HDMITX_EN_TX_POSDIV**	RG_HDMITX_EN_TX_POSDIV	**enable TX /5 circuit** 1'b0: disable 1'b1: enable
0	**RG_HDMITX_EN_TX_CKLDO**	RG_HDMITX_EN_TX_CKLDO	**enable TX half-rate clock LDO** 1'b0: disable 1'b1: enable

1020910C HDMI_CON3 HDMI control register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HDMITX_TEST_DIV_SEL		RG_HDMITX_TEST_SEL_TOP		RG_HDMITX_TEST_EN				RG_HDMITX_SER_BIST_TOG_CKCH	RG_HDMITX_SER_EIN_SEL_CKCH	RG_HDMITX_TEST_SEL					
Type	RW		RW		RW				RW	RW	RW					
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Mnemonic	Name	Description
31:30	**RG_HDMITX_TEST_DIV_SEL**	RG_HDMITX_TEST_DIV_SEL	
29:28	**RG_HDMITX_TEST_SEL_TOP**	RG_HDMITX_TEST_SEL_TOP	
27:24	**RG_HDMITX_TEST_EN**	RG_HDMITX_TEST_EN	
23	**RG_HDMITX_SER_BIST_TOG_CKCH**	RG_HDMITX_SER_BIST_TOG_CKCH	
22	**RG_HDMITX_SER_EIN_SEL_CKCH**	RG_HDMITX_SER_EIN_SEL_CKCH	
21:16	**RG_HDMITX_TEST_SEL**	RG_HDMITX_TEST_SEL	

10209110 HDMI_CON4 HDMI control register 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HDMITX_RESERVE[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_HDMITX_RESERVE[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RG_HDMITX_RESERVE**	RG_HDMITX_RESERVE	**reserved**

10209114 HDMI_CON5 HDMI control register 5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RGS_HDMITX_PL				RGS_HDMITX_ABIST_51EDG				RGS_HDMITX_ABIST_51LEV				RGS_HDMITX_ABIST_21EDG			

	UG_TST															
Type	RO				RO				RO				RO			
Reset	0				0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RGS_HDMITX_ABIST_21LEV				RGS_HDMITX_CAL_STATUS											
Type	RO				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0				

Bit(s)	Mnemonic	Name	Description
31	RGS_HDMITX_PLUG_TST	RGS_HDMITX_PLUG_TST	**Plug test result** 0: RX disappear 1: RX plug in
27:24	RGS_HDMITX_ABIST_51EDG	RGS_HDMITX_ABIST_51EDG	**ABIST 5T1SER edge test result**
23:20	RGS_HDMITX_ABIST_51LEV	RGS_HDMITX_ABIST_51LEV	**ABIST 5T1SER level test result**
19:16	RGS_HDMITX_ABIST_21EDG	RGS_HDMITX_ABIST_21EDG	**ABIST 2T1SER edge test result**
15:12	RGS_HDMITX_ABIST_21LEV	RGS_HDMITX_ABIST_21LEV	**ABIST 2T1SER level test result**
11:4	RGS_HDMITX_CAL_STATUS	RGS_HDMITX_CAL_STATUS	**Calibration status** 0: fail 1: success

10209118 HDMI_CON6 HDMI control register 6 09040000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HTPLL_EN	RG_HTPLL_FBKDIV							RG_HTPLL_DDSFBK_EN	RG_HTPLL_RLH_EN	RG_HTPLL_FBKSEL		RG_HTPLL_PREDIV		RG_HTPLL_POSDIV	
Type	RW	RW							RW	RW	RW		RW		RW	
Reset	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_HTPLL_IC				RG_HTPLL_IR				RG_HTPLL_BP				RG_HTPLL_BC		RG_HTPLL_BR	
Type	RW				RW				RW				RW		RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	RG_HTPLL_EN	RG_HTPLL_EN	**PLL Enable** 1'b0: Disable 1'b1: Enable
30:24	RG_HTPLL_FBKDIV	RG_HTPLL_FBKDIV	**Feedback divide ratio** 7'd0: /1 7'd1: /2 7'd127: /128
23	RG_HTPLL_DDSFBK_E	RG_HTPLL_DDSFBK_EN	**DDS Feedback Enable** 1'b0: Disable

Bit(s)	Mnemonic	Name	Description
	N		1'b1: Enable
22	**RG_HTPLL_RLH_EN**	RG_HTPLL_RLH_EN	**Feedback Relatch Enable** 1'b0: Disable 1'b1: Enable
21:20	**RG_HTPLL_FBKSEL**	RG_HTPLL_FBKSEL	**Feedback clock select** 2'b00: VCO/1 2'b01: VCO/2 2'b1X: VCO/4
19:18	**RG_HTPLL_PREDIV**	RG_HTPLL_PREDIV	**Pre-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4
17:16	**RG_HTPLL_POSDIV**	RG_HTPLL_POSDIV	**Post-divider ratio** 2'b00: /1 2'b01: /2 2'b10: /4 2'b11: Gating
15:12	**RG_HTPLL_IC**	RG_HTPLL_IC	**I-path current adjustment** MSB 5uA 5uA 2.5uA LSB 1.25uA
11:8	**RG_HTPLL_IR**	RG_HTPLL_IR	**P-path current adjustment** MSB 40uA 20uA 10uA LSB 5uA
7:4	**RG_HTPLL_BP**	RG_HTPLL_BP	**P-path capacitance adjustment** MSB=1pF 0.5pF 0.25pF LSB=125fF
3:2	**RG_HTPLL_BC**	RG_HTPLL_BC	**I-path capacitance adjustment** 0:1pF 1:2pF 2:3pF 3:4pF
1:0	**RG_HTPLL_BR**	RG_HTPLL_BR	**P-path resistance adjustment** 2'b00:60kohm 2'b01:40kohm 2'b10:20kohm 2'b11:10kohm

1020911C HDMI_CON7 HDMI control register 7 8080000F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_HTPLL_AUTOK_LOAD	RG_HTPLL_DIVEN			RG_HTPLL_AUTOK_KS		RG_HTPLL_AUTOK_KF		RG_HTPLL_AUTOK_EN	RG_HTPLL_BAND						
Type	RW	RW			RW		RW		RW	RW						
Reset	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_H	RG_H	RG_H	RG_H	RG_H	RG_H	RG_H	RG_H	RG_HTPLL_RESERVE							

	TPLL_MONCK_EN	TPLL_MONVC_EN	TPLL_MONREF_EN	TPLL_VOD_EN	TPLL_DET_EN	TPLL_OSC_RST	TPLL_BIAS_LPF_EN	TPLL_BIAS_EN								
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31	**RG_HTPLL_AUTOK_LOAD**	RG_HTPLL_AUTOK_LOAD	**Load Last-Time Auto K-Band Result** 1'b0: Load 1'b1: Unload
30:28	**RG_HTPLL_DIVEN**	RG_HTPLL_DIVEN	**Time domain cap multiplication ratio** 3'd0: x1 3'd1: x2 3'd2: x4 ... 3'd6: x64
27:26	**RG_HTPLL_AUTOK_KS**	RG_HTPLL_AUTOK_KS	**Auto K-Band Time Control**
25:24	**RG_HTPLL_AUTOK_KF**	RG_HTPLL_AUTOK_KF	**Auto K-Band Time Control**
23	**RG_HTPLL_AUTOK_EN**	RG_HTPLL_AUTOK_EN	**Auto K-Band Enable** 1'b0: Disable 1'b1: Enable
22:16	**RG_HTPLL_BAND**	RG_HTPLL_BAND	**Manual PLL Band Selection** 6'b000001: Lowest Band 6'b111110: Highest Band
15	**RG_HTPLL_MONCK_EN**	RG_HTPLL_MONCK_EN	**Monitor clock Enable** 1'b0: Disable 1'b1: Enable
14	**RG_HTPLL_MONVC_EN**	RG_HTPLL_MONVC_EN	**Monitor Vctrl Enable** 1'b0: Disable 1'b1: Enable
13	**RG_HTPLL_MONREF_EN**	RG_HTPLL_MONREF_EN	**Monitor reference Enable** 1'b0: Disable 1'b1: Enable
12	**RG_HTPLL_VOD_EN**	RG_HTPLL_VOD_EN	**CHP OverDrive Enable** (If AVDD12 > DVDD10 & use DVDD10 as LV reference TIE High, else TIE Low) 1'b0: Disable 1'b1: Enable
11	**RG_HTPLL_DET_EN**	RG_HTPLL_DET_EN	**PLL WatchDog Enable** 1'b0: Disable 1'b1: Enable
10	**RG_HTPLL_OSC_RST**	RG_HTPLL_OSC_RST	**Reset WatchDog Flag** 1'b0: Normal 1'b1: Reset
9	**RG_HTPLL_BIAS_LPF_EN**	RG_HTPLL_BIAS_LPF_EN	**Constant-Gm Bias LPF Enable** 1'b0: Bypass LPF 1'b1: LPF Enable
8	**RG_HTPLL_BIAS_EN**	RG_HTPLL_BIAS_EN	**Constant-Gm Bias Enable** 1'b0:Disable 1'b1: Enable
7:0	**RG_HTPLL_**	RG_HTPLL_RESE	**Reserve**

Bit(s)	Mnemonic	Name	Description
	RESERVE	RVE	

10209120 HDMI_CON8 HDMI control register 8 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RGS_HTPLL_AUTOK_FAIL								RGS_HTPLL_AUTOK_BAND							
Type	RO								RO							
Reset	0	0	0	0	0	0	0		0							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RGS_HTPLL_AUTOK_PASS															
Type	RO															
Reset	0															

Bit(s)	Mnemonic	Name	Description
31:25	**RGS_HTPLL_AUTOK_FAIL**	RGS_HTPLL_AUTOK_FAIL	**ABIST 5T1SER level test result**
23	**RGS_HTPLL_AUTOK_BAND**	RGS_HTPLL_AUTOK_BAND	**Plug test result** 0: RX disappear 1: RX plug in
15	**RGS_HTPLL_AUTOK_PASS**	RGS_HTPLL_AUTOK_PASS	**ABIST 5T1SER edge test result**

10209200 ARMPLL_CON0 ARMPLL Control Register 0 00000100

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								ARMPLL_SDM_FRA_EN								ARMPLL_EN
Type								RW								RW
Reset								1								0

Bit(s)	Mnemonic	Name	Description
8	**ARMPLL_SDM_FRA_EN**	ARMPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
0	**ARMPLL_EN**	ARMPLL_EN	**Enables PLL**

10209204 ARMPLL_CON1 ARMPLL Control Register 1 800A8000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARMPLL_SDM_PCW_CHG					ARMPLL_POSDIV						ARMPLL_SDM_PCW[20:16]				
Type	RW					RW						RW				
Reset	1					0	0	0				0	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ARMPLL_SDM_PCW[15:0]															
Type	RW															
Reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**ARMPLL_SDM_PCW_CHG**	ARMPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
26:24	**ARMPLL_POSDIV**	ARMPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
20:0	**ARMPLL_SDM_PCW**	ARMPLL_SDM_PCW	**Feedback divide ratio**

1020920C ARMPLL_PWR_CON0 ARMPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARMPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ARMPLL_SDM_ISO_EN	ARMPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**ARMPLL_SDM_PWR_ACK**	ARMPLL_SDM_PWR_ACK	**ARMPLL power ack**
1	**ARMPLL_SDM_ISO_EN**	ARMPLL_SDM_ISO_EN	**Enables ARMPLL iso**
0	**ARMPLL_SDM_PWR_ON**	ARMPLL_SDM_PWR_ON	**ARMPLL power-on**

10209210 MAINPLL_CON0 MAINPLL Control Register 0 78000110

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		MAINPLL_DIV2_EN	MAINPLL_DIV3_EN	MAINPLL_DIV5_EN	MAINPLL_DIV7_EN			MAINPLL_DIV_RSTB								
Type		RW	RW	RW	RW			RW								
Reset		1	1	1	1			0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								MAINPLL_SDM_FRA_EN		MAINPLL_POSDIV						MAINPLL_EN
Type								RW		RW						RW
Reset								1		0	0	1				0

Bit(s)	Mnemonic	Name	Description
30	**MAINPLL_DIV2_EN**	MAINPLL_DIV2_EN	**Enables MAINPLL DIV2**
29	**MAINPLL_DIV3_EN**	MAINPLL_DIV3_EN	**Enables MAINPLL DIV3**
28	**MAINPLL_DIV5_EN**	MAINPLL_DIV5_EN	**Enables MAINPLL DIV5**
27	**MAINPLL_DIV7_EN**	MAINPLL_DIV7_EN	**Enables MAINPLL DIV7**
24	**MAINPLL_DIV_RSTB**	MAINPLL_DIV_RSTB	**PLL divider reset bar** 0: Reset 1: Enable
8	**MAINPLL_SDM_FRA_EN**	MAINPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**MAINPLL_POSDIV**	MAINPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**MAINPLL_EN**	MAINPLL_EN	**Enables PLL**

10209214 MAINPLL_CON1 MAINPLL Control Register 1 80150000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINPLL_SDM_PCW_CHG											MAINPLL_SDM_PCW[20:16]				
Type	RW											RW				
Reset	1											1	0	1	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAINPLL_SDM_PCW[15:0]															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**MAINPLL_SDM_PCW_CHG**	MAINPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:0	**MAINPLL_SDM_PCW**	MAINPLL_SDM_PCW	**Feedback divide ratio**

1020921C **MAINPLL_PWR_CON0** **MAINPLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															MAINPLL_SDM_ISO_EN	MAINPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**MAINPLL_SDM_PWR_ACK**	MAINPLL_SDM_PWR_ACK	**MAINPLL power ack**
1	**MAINPLL_SDM_ISO_EN**	MAINPLL_SDM_ISO_EN	**Enables MAINPLL iso**
0	**MAINPLL_SDM_PWR_ON**	MAINPLL_SDM_PWR_ON	**MAINPLL power-on**

10209220 **UNIVPLL_CON0** **UNIVPLL Control Register 0** **FC000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	UNIVPLL_DIV2_EN	UNIVPLL_DIV3_EN	UNIVPLL_DIV5_EN	UNIVPLL_DIV7_EN	UNIV48M_EN	USB48M_EN		UNIVPLL_DIV_RSTB								
Type	RW	RW	RW	RW	RW	RW		RW								
Reset	1	1	1	1	1	1		0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								UNIVPLL_SDM_FRA_EN		UNIVPLL_POSDIV						UNIVPLL_EN
Type								RW		RW						RW
Reset								0		0	0	0				0

Bit(s)	Mnemonic	Name	Description
31	**UNIVPLL_DIV2_EN**	UNIVPLL_DIV2_EN	**Enables UNIVPLL DIV2**
30	**UNIVPLL_DIV3_EN**	UNIVPLL_DIV3_EN	**Enables UNIVPLL DIV3**
29	**UNIVPLL_DIV5_EN**	UNIVPLL_DIV5_EN	**Enables UNIVPLL DIV5**
28	**UNIVPLL_DIV7_EN**	UNIVPLL_DIV7_EN	**Enables UNIVPLL DIV7**
27	**UNIV48M_EN**	UNIV48M_EN	**Enables UNIV 48M clock**
26	**USB48M_EN**	USB48M_EN	**Enables USB 48M clock**
24	**UNIVPLL_DIV_RSTB**	UNIVPLL_DIV_RSTB	**PLL divider reset bar** 0: Reset 1: Enable
8	**UNIVPLL_SDM_FRA_EN**	UNIVPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**UNIVPLL_POSDIV**	UNIVPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**UNIVPLL_EN**	UNIVPLL_EN	**Enables PLL**

10209224 UNIVPLL_CON1 UNIVPLL Control Register 1 800C0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	UNIVPLL_SDM_PCW_CHG											UNIVPLL_SDM_PCW[6:2]				
Type	RW											RW				
Reset	1											0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	UNIVPLL_SDM_PCW[1:0]															
Type	RW															
Reset	0	0														

Bit(s)	Mnemonic	Name	Description
31	**UNIVPLL_SDM_PCW_CHG**	UNIVPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:14	**UNIVPLL_SDM_PCW**	UNIVPLL_SDM_PCW	**Feedback divide ratio**

1020922C UNIVPLL_PWR_CON0 UNIVPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	UNIVPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															UNIVPLL_SDM_ISO_EN	UNIVPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**UNIVPLL_SDM_PWR_ACK**	UNIVPLL_SDM_PWR_ACK	**UNIVPLL power ack**
1	**UNIVPLL_SDM_ISO_EN**	UNIVPLL_SDM_ISO_EN	**Enables UNIVPLL iso**
0	**UNIVPLL_SDM_PWR_ON**	UNIVPLL_SDM_PWR_ON	**UNIVPLL power-on**

10209230 MMPLL_CON0 MMPLL Control Register 0 00000120

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								MMPLL_SDM_FRA_EN		MMPLL_POSDIV						MMPLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	**MMPLL_SDM_FRA_EN**	MMPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**MMPLL_POSDIV**	MMPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**MMPLL_EN**	MMPLL_EN	**Enables PLL**

10209234 MMPLL_CON1 MMPLL Control Register 1 80134000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMPLL_SDM_PCW_CHG											MMPLL_SDM_PCW[20:16]				
Type	RW											RW				
Reset	1											1	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MMPLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**MMPLL_SDM_PCW_CHG**	MMPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:0	**MMPLL_SDM_PCW**	MMPLL_SDM_PCW	**Feedback divide ratio**

1020923C **MMPLL_PWR_CON0** **MMPLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															MMPLL_SDM_ISO_EN	MMPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**MMPLL_SDM_PWR_ACK**	MMPLL_SDM_PWR_ACK	**MMPLL power ack**
1	**MMPLL_SDM_ISO_EN**	MMPLL_SDM_ISO_EN	**Enables MMPLL iso**
0	**MMPLL_SDM_PWR_ON**	MMPLL_SDM_PWR_ON	**MMPLL power-on**

10209240 **MSDCPLL_CON0** **MSDCPLL Control Register 0** **00000120**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name								MSDCPLL_SDM_FRA_EN		MSDCPLL_POSDIV						MSDCPLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	**MSDCPLL_SDM_FRA_EN**	MSDCPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**MSDCPLL_POSDIV**	MSDCPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**MSDCPLL_EN**	MSDCPLL_EN	**Enables PLL**

10209244 MSDCPLL_CON1 MSDCPLL Control Register 1 800F6276

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSDCPLL_SDM_PCW_CHG											MSDCPLL_SDM_PCW[20:16]				
Type	RW											RW				
Reset	1											0	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDCPLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	1	0	0	0	1	0	0	1	1	1	0	1	1	0

Bit(s)	Mnemonic	Name	Description
31	**MSDCPLL_SDM_PCW_CHG**	MSDCPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:0	**MSDCPLL_SDM_PCW**	MSDCPLL_SDM_PCW	**Feedback divide ratio**

1020924C MSDCPLL_PWR_CON0 MSDCPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSDCPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															MSDCPLL_SDM_ISO_EN	MSDCPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**MSDCPLL_SDM_PWR_ACK**	MSDCPLL_SDM_PWR_ACK	**MSDCPLL power ack**
1	**MSDCPLL_SDM_ISO_EN**	MSDCPLL_SDM_ISO_EN	**Enables MSDCPLL iso**
0	**MSDCPLL_SDM_PWR_ON**	MSDCPLL_SDM_PWR_ON	**MSDCPLL power-on**

10209250 TVDPLL_CON0 TVDPLL Control Register 0 00000130

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								TVDPLL_SDM_FRA_EN		TVDPLL_POSDIV						TVDPLL_EN
Type								RW		RW						RW
Reset								1		0	1	1				0

Bit(s)	Mnemonic	Name	Description
8	**TVDPLL_SDM_FRA_EN**	TVDPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**TVDPLL_POSDIV**	TVDPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**TVDPLL_EN**	TVDPLL_EN	**Enables PLL**

10209254 TVDPLL_CON1 TVDPLL Control Register 1 800B6C4F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVDPLL_SDM_PCW_CHG											TVDPLL_SDM_PCW[20:16]				
Type	RW											RW				

Reset	1											0	1	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVDPLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	1	0	1	1	0	0	0	1	0	0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31	**TVDPLL_SDM_PCW_CHG**	TVDPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:0	**TVDPLL_SDM_PCW**	TVDPLL_SDM_PCW	**Feedback divide ratio**

1020925C **TVDPLL_PWR_CON0** **TVDPLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVDPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															TVDPLL_SDM_ISO_EN	TVDPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**TVDPLL_SDM_PWR_ACK**	TVDPLL_SDM_PWR_ACK	**TVDPLL power ack**
1	**TVDPLL_SDM_ISO_EN**	TVDPLL_SDM_ISO_EN	**Enables TVDPLL iso**
0	**TVDPLL_SDM_PWR_ON**	TVDPLL_SDM_PWR_ON	**TVDPLL power-on**

10209260 **TVDPLL_SSC_CON0** **TVDPLL SSC Control Register 0** **00020000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															TVDPLL_SDM_SSC_PH_INIT	TVDPLL_SDM_SSC_EN
Type															RW	RW
Reset															1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVDPLL_SDM_SSC_PRD															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
17		TVDPLL_SDM_SSC_PH_INIT	**SDM SSC phase initial** 1'b0: Upward 1'b1: Downward
16		TVDPLL_SDM_SSC_EN	**SDM SSC enable** 1'b0: Disable 1'b1: Enable
15:0		TVDPLL_SDM_SSC_PRD	**SDM SSC period** 16'd0: Min 16'd65536: Max

10209264 TVDPLL_SSC_CON1 TVDPLL SSC Control Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVDPLL_SDM_SSC_DELTA1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVDPLL_SDM_SSC_DELTA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		TVDPLL_SDM_SSC_DELTA1	**SDM SSC amplitude 1** 16'd0: Min 16'd65536: Max
15:0		TVDPLL_SDM_SSC_DELTA	**SDM SSC amplitude** 16'd0: Min 16'd65536: Max

10209270 AUD1PLL_CON0 AUD1PLL Control Register 0 00000140

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								AUD1PLL_SDM_FRA_EN		AUD1PLL_POSDIV						AUD1PLL_EN
Type								RW		RW						RW
Reset								1		1	0	0				0

Bit(s)	Mnemonic	Name	Description
8	AUD1PLL_SDM_FRA_EN	AUD1PLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode

Bit(s)	Mnemonic	Name	Description
			1: Fractional mode
6:4	**AUD1PLL_POSDIV**	AUD1PLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**AUD1PLL_EN**	AUD1PLL_EN	**Enables PLL**

10209274 AUD1PLL_CON1 AUD1PLL Control Register 1 BC7EA932

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AUD1PLL_SDM_PCW_CHG	AUD1PLL_SDM_PCW[30:16]														
Type	RW	RW														
Reset	1	0	1	1	1	1	0	0	0	1	1	1	1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUD1PLL_SDM_PCW[15:0]															
Type	RW															
Reset	1	0	1	0	1	0	0	1	0	0	1	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31	**AUD1PLL_SDM_PCW_CHG**	AUD1PLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
30:0	**AUD1PLL_SDM_PCW**	AUD1PLL_SDM_PCW	**Feedback divide ratio**

1020927C AUD1PLL_PWR_CON0 AUDPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AUDPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															AUD1PLL_SDM_ISO_EN	AUD1PLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31	**AUDPLL_SDM_PWR_ACK**	AUDPLL_SDM_PWR_ACK	**AUD1PLL power ack**
1	**AUD1PLL_SDM_ISO_EN**	AUD1PLL_SDM_ISO_EN	**Enables AUD1PLL iso**
0	**AUD1PLL_SDM_PWR_ON**	AUD1PLL_SDM_PWR_ON	**AUD1PLL power-on**

10209280 TRGPLL_CON0 TRGPLL Control Register 0 00000110

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								**TRGPLL_SDM_FRA_EN**		**TRGPLL_POSDIV**						**TRGPLL_EN**
Type								RW		RW						RW
Reset								1		0	0	1				0

Bit(s)	Mnemonic	Name	Description
8	**TRGPLL_SDM_FRA_EN**	TRGPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**TRGPLL_POSDIV**	TRGPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**TRGPLL_EN**	TRGPLL_EN	**Enables PLL**

10209284 TRGPLL_CON1 TRGPLL Control Register 1 B7C4EC4F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**TRGPLL_SDM_PCW_CHG**	**TRGPLL_SDM_PCW[30:16]**														
Type	RW	RW														
Reset	1	0	1	1	0	1	1	1	1	1	0	0	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**TRGPLL_SDM_PCW[15:0]**															
Type	RW															
Reset	1	1	1	0	1	1	0	0	0	1	0	0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31	**TRGPLL_SDM_PCW_CHG**	TRGPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**

Bit(s)	Mnemonic	Name	Description
	G		
30:0	TRGPLL_SDM_PCW	TRGPLL_SDM_PCW	Feedback divide ratio

1020928C TRGPLL_PWR_CON0 TRGPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TRGPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															TRGPLL_SDM_ISO_EN	TRGPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	TRGPLL_SDM_PWR_ACK	TRGPLL_SDM_PWR_ACK	TRGPLL power ack
1	TRGPLL_SDM_ISO_EN	TRGPLL_SDM_ISO_EN	Enables TRGPLL iso
0	TRGPLL_SDM_PWR_ON	TRGPLL_SDM_PWR_ON	TRGPLL power-on

10209290 ETHPLL_CON0 ETHPLL Control Register 0 00000120

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								ETHPLL_SDM_FRA_EN		ETHPLL_POSDIV						ETHPLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	ETHPLL_SDM_FRA_EN	ETHPLL_SDM_FRA_EN	Enables SDMPLL fractional mode 0: Integer mode 1: Fractional mode
6:4	ETHPLL_POSDIV	ETHPLL_POSDIV	Post divide ratio 000: /1 001: /2

Bit(s)	Mnemonic	Name	Description
			010: /4 011: /8 100: /16
0	**ETHPLL_EN**	ETHPLL_EN	**Enables PLL**

10209294 ETHPLL_CON1 ETHPLL Control Register 1 CCEC4EC5

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ETHPLL_SDM_PCW_CHG	ETHPLL_SDM_PCW[30:16]														
Type	RW	RW														
Reset	1	1	0	0	1	1	0	0	1	1	1	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ETHPLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	0	0	1	1	1	0	1	1	0	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
31	**ETHPLL_SDM_PCW_CHG**	ETHPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
30:0	**ETHPLL_SDM_PCW**	ETHPLL_SDM_PCW	**Feedback divide ratio**

1020929C ETHPLL_PWR_CON0 ETHPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ETHPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ETHPLL_SDM_ISO_EN	ETHPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**ETHPLL_SDM_PWR_ACK**	ETHPLL_SDM_PWR_ACK	**PLL power ack**
1	**ETHPLL_SDM_ISO_EN**	ETHPLL_SDM_ISO_EN	**Enables PLL iso**
0	**ETHPLL_SD**	ETHPLL_SDM_PW	**PLL power-on**

Bit(s)	Mnemonic	Name	Description
	M_PWR_ON	R_ON	

102092A0 **VDECPLL_CON0** **VDECPLL Control Register 0** **00000120**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								**VDECPLL_SDM_FRA_EN**		**VDECPLL_POSDIV**						**VDECPLL_EN**
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	**VDECPLL_SDM_FRA_EN**	VDECPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**VDECPLL_POSDIV**	VDECPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**VDECPLL_EN**	VDECPLL_EN	**Enables PLL**

102092A4 **VDECPLL_CON1** **VDECPLL Control Register 1** **B4000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**VDECPLL_SDM_PCW_CHG**	**VDECPLL_SDM_PCW[30:16]**														
Type	RW	RW														
Reset	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**VDECPLL_SDM_PCW[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**VDECPLL_SDM_PCW_CHG**	VDECPLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
30:0	**VDECPLL_SDM_PCW**	VDECPLL_SDM_PCW	**Feedback divide ratio**

102092AC VDECPLL_PWR_CON0 VDECPLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VDECPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															VDECPLL_SDM_ISO_EN	VDECPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	VDECPLL_SDM_PWR_ACK	VDECPLL_SDM_PWR_ACK	PLL power ack
1	VDECPLL_SDM_ISO_EN	VDECPLL_SDM_ISO_EN	Enables PLL iso
0	VDECPLL_SDM_PWR_ON	VDECPLL_SDM_PWR_ON	PLL power-on

102092B0 HADDS2PLL_CON0 HADDS2PLL Control Register 0 00000120

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								HADDS2PLL_SDM_FRA_EN		HADDS2PLL_POSDIV						HADDS2PLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	HADDS2PLL_SDM_FRA_EN	HADDS2PLL_SDM_FRA_EN	Enables SDMPLL fractional mode 0: Integer mode 1: Fractional mode
6:4	HADDS2PLL_POSDIV	HADDS2PLL_POSDIV	Post divide ratio 000: /1 001: /2 010: /4 011: /8

Bit(s)	Mnemonic	Name	Description
			100: /16
0	**HADDS2PLL_EN**	HADDS2PLL_EN	**Enables PLL**

102092B4 **HADDS2PLL_CON1** **HADDS2PLL Control Register 1** **AD5EFEE6**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**HADDS2PLL_SDM_PCW_CHG**	**HADDS2PLL_SDM_PCW[30:16]**														
Type	RW	RW														
Reset	1	0	1	0	1	1	0	1	0	1	0	1	1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**HADDS2PLL_SDM_PCW[15:0]**															
Type	RW															
Reset	1	1	1	1	1	1	1	0	1	1	1	0	0	1	1	0

Bit(s)	Mnemonic	Name	Description
31	**HADDS2PLL_SDM_PCW_CHG**	HADDS2PLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
30:0	**HADDS2PLL_SDM_PCW**	HADDS2PLL_SDM_PCW	**Feedback divide ratio**

102092BC **HADDS2PLL_PWR_CON0** **HADDS2PLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**HADDS2PLL_SDM_PWR_ACK**															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															**HADDS2PLL_SDM_ISO_EN**	**HADDS2PLL_SDM_PWR_ON**
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**HADDS2PLL_SDM_PWR_ACK**	HADDS2PLL_SDM_PWR_ACK	**PLL power ack**
1	**HADDS2PLL_SDM_ISO_EN**	HADDS2PLL_SDM_ISO_EN	**Enables PLL iso**

Bit(s)	Mnemonic	Name	Description
0	**HADDS2PLL_SDM_PWR_ON**	HADDS2PLL_SDM_PWR_ON	**PLL power-on**

102092C0 AUD2PLL_CON0 AUD2PLL Control Register 0 00000120

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								AUD2PLL_SDM_FRA_EN		AUD2PLL_POSDIV						AUD2PLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	**AUD2PLL_SDM_FRA_EN**	AUD2PLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**AUD2PLL_POSDIV**	AUD2PLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**AUD2PLL_EN**	AUD2PLL_EN	**Enables PLL**

102092C4 AUD2PLL_CON1 AUD2PLL Control Register 1 A9AF46FD

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AUD2PLL_SDM_PCW_CHG	AUD2PLL_SDM_PCW[30:16]														
Type	RW	RW														
Reset	1	0	1	0	1	0	0	1	1	0	1	0	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUD2PLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	0	0	0	1	1	0	1	1	1	1	1	1	0	1

Bit(s)	Mnemonic	Name	Description
31	**AUD2PLL_SDM_PCW_CHG**	AUD2PLL_SDM_PCW_CHG	**Feedback divide ratio update signal**

Bit(s)	Mnemonic	Name	Description
30:0	**AUD2PLL_SDM_PCW**	AUD2PLL_SDM_PCW	**Feedback divide ratio**

102092CC **AUD2PLL_PWR_CON0** **AUD2PLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AUD2PLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															AUD2PLL_SDM_ISO_EN	AUD2PLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**AUD2PLL_SDM_PWR_ACK**	AUD2PLL_SDM_PWR_ACK	**AUD2PLL power ack**
1	**AUD2PLL_SDM_ISO_EN**	AUD2PLL_SDM_ISO_EN	**Enables AUD2PLL iso**
0	**AUD2PLL_SDM_PWR_ON**	AUD2PLL_SDM_PWR_ON	**AUD2PLL power-on**

102092D0 **TVD2PLL_CON0** **TVD2PLL Control Register 0** **00000120**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								TVD2PLL_SDM_FRA_EN		TVD2PLL_POSDIV						TVD2PLL_EN
Type								RW		RW						RW
Reset								1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
8	**TVD2PLL_SDM_FRA_EN**	TVD2PLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**TVD2PLL_POSDIV**	TVD2PLL_POSDIV	**Post divide ratio** 000: /1

Bit(s)	Mnemonic	Name	Description
			001: /2 010: /4 011: /8 100: /16
0	**TVD2PLL_EN**	TVD2PLL_EN	**Enables PLL**

102092D4 TVD2PLL_CON1 TVD2PLL Control Register 1 800B6C4F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVD2PLL_SDM_PCW_CHG											TVD2PLL_SDM_PCW[20:16]				
Type	RW											RW				
Reset	1											0	1	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVD2PLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	1	1	0	1	1	0	0	0	1	0	0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31	**TVD2PLL_SDM_PCW_CHG**	TVD2PLL_SDM_PCW_CHG	**Feedback divide ratio update signal**
20:0	**TVD2PLL_SDM_PCW**	TVD2PLL_SDM_PCW	**Feedback divide ratio**

102092DC TVD2PLL_PWR_CON0 TVD2PLL Power Control Register 0 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVD2PLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															TVD2PLL_SDM_ISO_EN	TVD2PLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**TVD2PLL_SDM_PWR_ACK**	TVD2PLL_SDM_PWR_ACK	**TVD2PLL power ack**
1	**TVD2PLL_S**	TVD2PLL_SDM_IS	**Enables TVD2PLL iso**

Bit(s)	Mnemonic	Name	Description
	DM_ISO_EN	O_EN	
0	**TVD2PLL_SDM_PWR_ON**	TVD2PLL_SDM_PWR_ON	**TVD2PLL power-on**

102092F0 **TVD2PLL_SSC_CON0** **TVD2PLL SSC Control Register 0** **00020000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															TVD2PLL_SDM_SSC_PH_INIT	TVD2PLL_SDM_SSC_EN
Type															RW	RW
Reset															1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVD2PLL_SDM_SSC_PRD															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
17		TVD2PLL_SDM_SSC_PH_INIT	**SDM SSC phase initial** 1'b0: Upward 1'b1: Downward
16		TVD2PLL_SDM_SSC_EN	**SDM SSC enable** 1'b0: Disable 1'b1: Enable
15:0		TVD2PLL_SDM_SSC_PRD	**SDM SSC period** 16'd0: Min 16'd65536: Max

102092F4 **TVD2PLL_SSC_CON1** **TVD2PLL SSC Control Register 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TVD2PLL_SDM_SSC_DELTA1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TVD2PLL_SDM_SSC_DELTA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		TVD2PLL_SDM_SSC_DELTA1	**SDM SSC amplitude 1** 16'd0: Min 16'd65536: Max
15:0		TVD2PLL_SDM_SSC_DELTA	**SDM SSC amplitude** 16'd0: Min 16'd65536: Max

10209400 **AP_AUXADC_CON0** **AUXADC Control Register 0** **00008800**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AUXADC_RSV															
Type	RW															
Reset	0	0	0	0	0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUXADC_DUALY				AUXADC_DUALX							AUXADC_PIRQ_RES	AUXADC_CALI			
Type	RW				RW							RW	RW			
Reset	1	0	0	0	1	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**AUXADC_RSV**	AUXADC_RSV	**Reserved register**
15:12	**AUXADC_DUALY**	AUXADC_DUALY	**Reserved**
11:8	**AUXADC_DUALX**	AUXADC_DUALX	**Reserved**
4	**AUXADC_PIRQ_RES**	AUXADC_PIRQ_RES	**PENIRQ pull-high resistance** 0: 55K 1: 90K
3:0	**AUXADC_CALI**	AUXADC_CALI	**ADC core bias current calibration** 0000: 1X 0001: 1.25X 0010: 1.5X 0011: 1.75X 0100: 2X 1000: 1X 1001: 0.8X 1010: 0.67X 1011: 0.57X 1100: 0.5X

10209404 **AP_AUXADC_CON1** **AUXADC Control Register 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																AUXADC_RTP_DISABLE
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**AUXADC_RTP_DIS**	AUXADC_RTP_DIS	**Disables RTP**

Bit(s)	Mnemonic	Name	Description
	TP_DISABLE	ABLE	

1020940C **AP_AUXADC_CON2** **AUXADC Control Register 2** **00004000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														AUXADC_8BDAC_TEST	AUXADC_HDMIRX_TEST	AUXADC_BUF_PWDB
Type														RW	RW	RW
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUXADC_BUF_BUFMODE	AUXADC_CUR_SEL			AUXADC_FS	AUXADC_VDAC_TEST	AUXADC_GPI9_EN	AUXADC_GPI8_EN	AUXADC_GPI7_EN	AUXADC_GPI6_EN	AUXADC_GPI5_EN	AUXADC_GPI4_EN	AUXADC_GPI3_EN	AUXADC_GPI2_EN	AUXADC_GPI1_EN	AUXADC_GPI0_EN
Type	RW	RW			RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
18		AUXADC_8BDAC_TEST	
17		AUXADC_HDMIRX_TEST	
16		AUXADC_BUF_PWDB	
15		AUXADC_BUF_BUFMODE	
14:12		AUXADC_CUR_SEL	
11		AUXADC_FS	
10		AUXADC_VDAC_TEST	
9		AUXADC_GPI9_EN	**PAD_AUX_IN9 scan in function enable** 1: enalbe 0: disable
8		AUXADC_GPI8_EN	**PAD_AUX_IN8 scan in function enable** 1: enalbe 0: disable
7		AUXADC_GPI7_EN	**PAD_AUX_IN7 scan in function enable** 1: enalbe 0: disable
6		AUXADC_GPI6_EN	**PAD_AUX_IN6 scan in function enable** 1: enalbe 0: disable
5		AUXADC_GPI5_EN	**PAD_AUX_IN5 scan in function enable** 1: enalbe 0: disable
4		AUXADC_GPI4_EN	**PAD_AUX_IN4 scan in function enable**

Bit(s)	Mnemonic	Name	Description
			1: enalbe 0: disable
3		AUXADC_GPI3_EN	**PAD_AUX_IN3 scan in function enable** 1: enalbe 0: disable
2		AUXADC_GPI2_EN	**PAD_AUX_IN2 scan in function enable** 1: enalbe 0: disable
1		AUXADC_GPI1_EN	**PAD_AUX_IN1 scan in function enable** 1: enalbe 0: disable
0		AUXADC_GPI0_EN	**PAD_AUX_IN0 scan in function enable** 1: enalbe 0: disable

10209600 TS_CON0 Thermal Sensor Control Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BGR_RSV								BGR_TCTRL							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BGR_CTRL				BGR_RSEL				TS2AUXADC			BGR_UGB	BGR_BUFIN	BGR_FSETUP	BGR_UNCHOP_PH	BGR_UNCHOP
Type	RW				RW				RW			RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**BGR_RSV**	BGR_RSV	**RG_BGR_RSV[7:1]: Reserved register** RG_BGR_RSV[0]: Enable VBGR output to PAD_TESTIIN 1: Enable 0: Disable
23:16	**BGR_TCTRL**	BGR_TCTRL	**BGR_TCTRL[1:0]: BGR Replica VBE TSensor chopping control** 00: Unchop, PH = 0 01: Unchop, PH = 1 10: Chop 11: Disable The current for VBE TSensor is also as the reference of calibrating VBE TSensor for 00/01/10 mode. By setting RG_BGR_CTRL[3]=1 and RG_BGR_TCTRL[1:0]=11,the current source of calibrating VBE can be applied externally (drawn from PD_TESTIIN). BGR_TCTRL[3:2]: PAD_TSOUT output selection 00: Disable TSOUT 01: IPTAT_OUT (for VBE calibration) 10: VBE_MCU 11: VBE_ABB RG_BGR_TCTRL[7:4]: IPTAT VBE TSensor current selection 0000: 100000000 for 1I calibrating VBE 0001: 010000000 for 1I calibrating VBE 0010: 001000000 for 1I calibrating VBE 0011: 000100000 for 1I calibrating VBE

Bit(s)	Mnemonic	Name	Description
			0100: 000010000 for 1I calibrating VBE 0101: 000001000 for 1I calibrating VBE 0110: 000000100 for 1I calibrating VBE 0111: 000000010 for 1I calibrating VBE 1000: 000000001 for 1I calibrating VBE 1001: 111000000 for 3I calibrating VBE 1010: 000111000 for 3I calibrating VBE 1011: 000000111 for 3I calibrating VBE 1100: 111111111 for 9I calibrating VBE
15:12	**BGR_CTRL**	BGR_CTRL	**RG_BGR_CTRL[2:0]: BGR resistor selection for denomerator** RG_BGR_CTRL[3]: Enable test current input 1: Enable test current. In this mode, PAD_TESTIIN is used for applying calibrating VBE TSensor current. (see RG_BGR_TCTRL[1:0].) 0: Disable test current
11:8	**BGR_RSEL**	BGR_RSEL	**BGR resistor selection for numerator**
7:6	**TS2AUXADC**	TS2AUXADC	**Enables output buffer and selects TS output to AUXADC** 00: Buffer on, TSMCU to AUXADC 01: Buffer on, TSABB to AUXADC 10: Buffer on, VBGR to AUXADC 11: Buffer off
4	**BGR_UGB**	BGR_UGB	**BGR VBUFFER force unity gain mode** 0: Disable 1: Enable
3	**BGR_BUFIN**	BGR_BUFIN	**BGR VBUFFER external input mode** 0: Disable 1: Enable
2	**BGR_FSETUP**	BGR_FSETUP	**BGR fast setup control** 0: Disable resistor in low pass filter, speed up settling 1: Enable low pass filter
1	**BGR_UNCHOP_PH**	BGR_UNCHOP_PH	**Selects BGR unchop mode phase**
0	**BGR_UNCHOP**	BGR_UNCHOP	**BGR unchop mode** 0: Chop 1: Unchop

10209604 TS_CON1 Thermal Sensor Control Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														**VBE_SEL**		
Type														RW		
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2:0	**VBE_SEL**	VBE_SEL	**VBE selection**

1000F800 VENCPLL_CON VENCPLL Control Register 0 00000120

0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					VENCPLL_MONVC_EN	VENCPLL_MONCK_EN	VENCPLL_LVROD_EN	VENCPLL_SDM_FRA_EN		VENCPLL_POSDIV						VENCPLL_EN
Type					RW	RW	RW	RW		RW						RW
Reset					0	0	0	1		0	1	0				0

Bit(s)	Mnemonic	Name	Description
11	**VENCPLL_MONVC_EN**	VENCPLL_MONVC_EN	
10		VENCPLL_MONCK_EN	
9		VENCPLL_LVROD_EN	
8	**VENCPLL_SDM_FRA_EN**	VENCPLL_SDM_FRA_EN	**Enables SDMPLL fractional mode** 0: Integer mode 1: Fractional mode
6:4	**VENCPLL_POSDIV**	VENCPLL_POSDIV	**Post divide ratio** 000: /1 001: /2 010: /4 011: /8 100: /16
0	**VENCPLL_EN**	VENCPLL_EN	**Enables PLL**

1000F804 **VENCPLL_CON1** **VENCPLL Control Register 1** **800B6000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VENCPLL_N_INFO_CHG											VENCPLL_N_INFO[20:16]				
Type	RW											RW				
Reset	1											0	1	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VENCPLL_N_INFO[15:0]															
Type	RW															
Reset	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**VENCPLL_N_INFO_CHG**	VENCPLL_N_INFO_CHG	**Feedback divide ratio update signal**
20:0	**VENCPLL_N_INFO**	VENCPLL_N_INFO	**Feedback divide ratio**

1000F80C **VENCPLL_PWR_CON0** **VENCPLL Power Control Register 0** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	VENCPLL_SDM_PWR_ACK															
Type	RO															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															VENCPLL_SDM_ISO_EN	VENCPLL_SDM_PWR_ON
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
31	**VENCPLL_SDM_PWR_ACK**	VENCPLL_SDM_PWR_ACK	**VENCPLL power ack**
1	**VENCPLL_SDM_ISO_EN**	VENCPLL_SDM_ISO_EN	**Enables VENCPLL iso**
0	**VENCPLL_SDM_PWR_ON**	VENCPLL_SDM_PWR_ON	**VENCPLL power-on**

3 Top Reset Generate Unit

3.1 Introduction

The top reset generator unit (TOPRGU) generates reset signals and distributes to each system. A watchdog timer is also included in this module.

3.2 Feature list

- Hardware reset signals for the whole chip
- Software controllable reset for each system (except for infrastructure and apmixedsys system)
- Watchdog timer
- Reset output signals for companion chips

3.3 Block Diagram

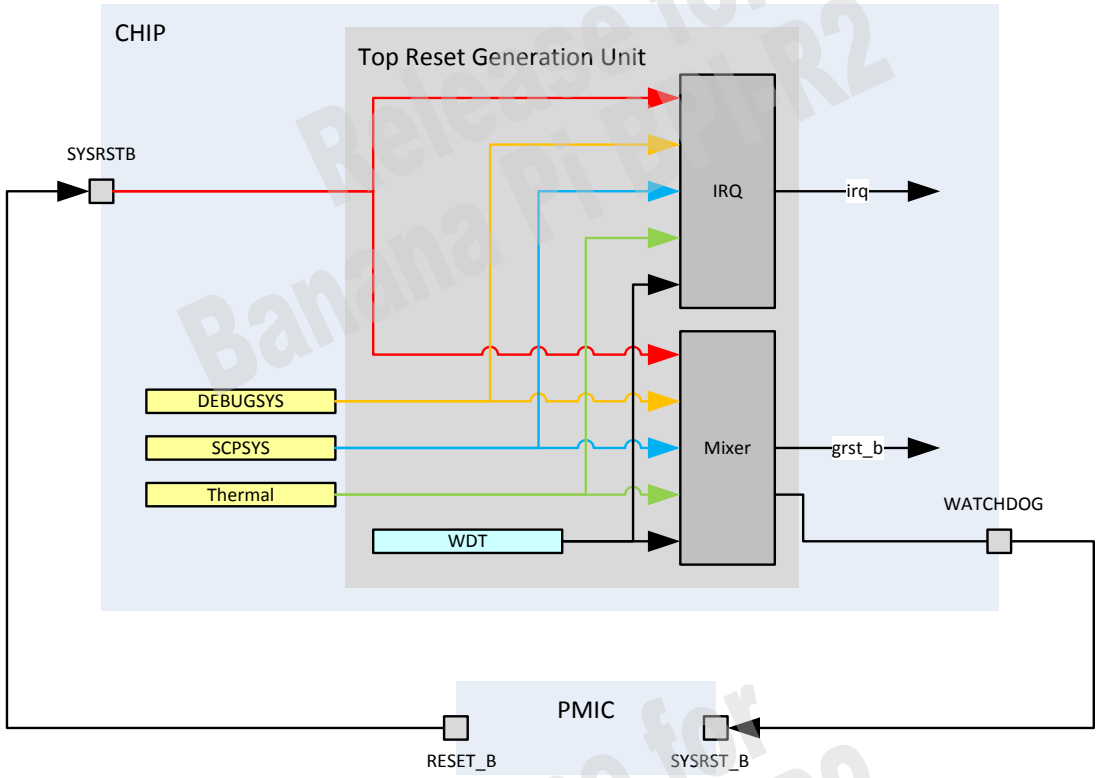


Figure 3-1: Block diagram of top reset generation unit

3.4 Register Definition

Module name: TOPRGU Base address: (+10007000h)

Address	Name	Width	Register function
10007000	WDT_MODE	32	Watchdog mode register (will not be reset)
10007004	WDT_LENGTH	32	Watchdog counter setting register
10007008	WDT_RESTART	32	Watchdog counter restart register
1000700C	WDT_STA	32	Watchdog status register (will not be reset)
10007010	WDT_INTERVAL	32	Watchdog reset pulse width register
10007014	WDT_SWRST	32	Software watchdog reset register
10007018	WDT_SWSYSRST	32	System software reset register
10007030	WDT_REQ_MODE	32	Reset request mode register (will not be reset)
10007034	WDT_REQ_IRQ_EN	32	Reset request IRQ enable register (will not be reset)
10007040	WDT_DEBUG_CTL	32	Debug control register (will not be reset)
10007050	WDT_INTERCORE_SYNC	32	Register for intercore sync
10007054	WDT_INTERCORE_SYNC_SET	32	Set control of register for intercore sync
10007058	WDT_INTERCORE_SYNC_CLR	32	Clear control of register for intercore sync

Module name: TOPRGU_2ND Base address: (+10000000h)

Address	Name	Width	Register function
10000310	RESET_DEGLITCH_KEY	32	Reset deglitch enable key register

10007000 WDT_MODE Watchdog Mode Register (will not be reset) 2200004D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	unlock_key															
Type	WO															
Reset	0	0	1	0	0	0	1	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									ddr_reserve_mode	dual_mode	irq_lvl_en		wdt_irq	exten	extpol	wdt_en
Type									RW	RW	RW		RW	RW	RW	RW
Reset									0	1	0		1	1	0	1

Bit(s)	Name	Description
31:24	unlock_key	**Write 0x22 to unlock the write protection of this register**
7	ddr_reserve_mode	**Enables ddr_reserve_mode** 0: Disable 1: Enable
6	dual_mode	**Enables dual_mode** Turn on watchdog timer and enable the correspondent irq_en and wdt_en if

Bit(s)	Name	Description
		dual_mode is used. 0: Disable 1: Enable
5	irq_lvl_en	**Selects IRQ type** 0: Edge (32K) 1: Level
3	wdt_irq	**Enables watchdog timer IRQ** 0: Trigger reset 1: Trigger IRQ
2	exten	**Enables watchdog output reset signal** 0: Disable 1: Enable
1	extpol	**Watchdog output reset signal polarity** 0: Active low 1: Active high
0	wdt_en	**Enables watchdog timer** 0: Disable 1: Enable

10007004 WDT_LENGTH Watchdog Counter Setting Register 0000FFE8

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	wdt_length											unlock_key				
Type	RW											WO				
Reset	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0

Bit(s)	Name	Description
15:5	wdt_length	**Watchdog time-out counter setting** The counter is restarted with {wdt_length [10:0], 1_1111_1111b}, and therefore the watchdog timer time-out period is multiple of 512*T32k=15.6ms.
4:0	unlock_key	**Write 0x8 to unlock the write protection of this register**

10007008 WDT_RESTART Watchdog Counter Restart Register 00001971

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	wdt_restart															
Type	WO															
Reset	0	0	0	1	1	0	0	1	0	1	1	1	0	0	0	1

Bit(s)	Name	Description
15:0	wdt_restart	Write 0x1971 to reset the watchdog time-out counter

1000700C WDT_STA Watchdog Status Register (will not be reset) 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	hw_wdt_rst	sw_wdt_rst	irq_assert										debug_rst			
Type	RO	RO	RO										RO			
Reset	0	0	0										0			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															spm_wdt_rst	thermal_rst
Type															RO	RO
Reset															0	0

	Name	Description
3	hw_wdt_rst	Indicates hardware watchdog generated reset is asserted
2	sw_wdt_rst	Indicates software watchdog generated reset is asserted
1	irq_assert	Indicates IRQ is asserted instead of reset
0	debug_rst	Indicates debug generated reset is asserted
1	spm_wdt_rst	Indicates scpsys time-out generated reset is asserted
0	thermal_rst	Indicates thermal generated by scpsys reset is asserted

10007010 WDT_INTERVAL Watchdog Reset Pulse Width Register 00000FFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					wdt_reset_interval											
Type					RW											
Reset					1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
11:0	wdt_reset_interval	**Resets pulse width generated by watchdog** Needs 2T 32K for TD modem. Unit: 1T=1x32kHz

10007014 WDT_SWRST Software Watchdog Reset Register 00001209

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	unlock_key															
Type	WO															
Reset	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	1

Bit(s)	Name	Description
15:0	unlock_key	Write 0x1209 to generate a software watchdog reset

10007018 WDT_SWSYSRST System Software Reset Register 88000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	unlock_key															
Type	WO															
Reset	1	0	0	0	1	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			bdp_disp_rst	conn_mcu_rst	hifsys_rst	apmixed_rst	conn_rst	infra_ao_rst	md_rst	ddrphy_rst	venc_img_rst	vdec_rst	ethdmasys_rst	mfg_rst	mm_rst	infra_rst
Type			RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	unlock_key	**Write 0x88 to unlock the write protection of this register.**
13	bdp_disp_rst	**Write 1 to reset BDP_DISP**
12	conn_mcu_rst	**Write 1 to reset CONN_MCU.**
11	hifsys_rst	**Write 1 to reset hifsys**
10	apmixed_rst	**Write 1 to reset APMIXEDSYS.**
9	conn_rst	**Write 1 to reset CONNSYS**
8	infra_ao_rst	**Write 1 to reset INFRA_AO**
7	md_rst	**Write 1 to reset the MODEM system.** Needs 2T 32K duration for TD modem.
6	ddrphy_rst	**Write 1 to reset DDRPHY and MEMPLL**
5	venc_img_rst	**Write 1 to reset VENC and IMG and its related pad macro (CAM, MIPI_RX)**
4	vdec_rst	**Write 1 to reset VDEC**
3	ethdma_rst	**Write 1 to reset ETHDMA**
2	mfg_rst	**Write 1 to reset MFG**
1	mm_rst	**Write 1 to reset MM and its related pad macro (SPI, DPI, MIPI_CFG, MIPI_TX)**
0	infra_rst	**Write 1 to reset INFRASYS and its related pad macro (NLI,EFUSE)**

10007030 WDT_REQ_MODE Reset Request Mode Register (will npt be reset) 33080003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	unlock_key												debug_en			
Type	WO												RW			
Reset	0	0	1	1	0	0	1	1					1			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name															scpsys_en	thermal_en
Type															RW	RW
Reset															1	1

Bit(s)	Name	Description
31:24	unlock_key	**Write 0x33 to unlock the write protection of this register**
19	debug_en	**Enables debugsys reset** 0: Disable 1: Enable
1	scpsys_en	**Enables scpsys reset** 0: Disable 1: Enable
0	thermal_en	**Enables thermal reset** 0: Disable 1: Enable

10007034 WDT_REQ_IRQ_EN Reset Request IRQ Enable Register (will not be reset) 44080003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	unlock_key												debug_irq			
Type	WO												RW			
Reset	0	1	0	0	0	1	0	0					1			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															scpsys_irq	thermal_irq
Type															RW	RW
Reset															1	1

Bit(s)	Name	Description
31:24	unlock_key	**Write 0x44 to unlock the write protection of this register**
19	debug_irq	**Triggers IRQ instead of reset when debugsys reset is enabled** 0: Trigger reset 1: Trigger IRQ
1	scpsys_irq	**Triggers IRQ instead of reset when scpsys reset is enabled** 0: Trigger reset 1: Trigger irq
0	thermal_irq	**Triggers IRQ instead of reset when thermal reset is enabled** 0: Trigger reset 1: Trigger IRQ

10007040 WDT_DEBUG_CTL Debug Control Register (will not be reset) FFFE00F1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	unlock_key														ddr_sref_sta	ddr_reserve_sta
Type	WO														RU	RO
Reset	0	1	0	1	1	0	0	1							z	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						rg_dramc_conf_iso	rg_dramc_iso	rg_dramc_sref	rg_dramc_timeout						rg_mcu_lath_en	rg_ddr_protect_en
Type						RW	RW	RW	RW						RW	RW
Reset						0	0	0	1	1	1	1			0	1

Bit(s)	Name	Description
31:24	unlock_key	**Write 0x59 to unlock the write protection of this register**
17	ddr_sref_sta	**Indicates the current DDR status** 0: Currently not in self-refresh mode 1: Currently in self-refresh mode
16	ddr_reserve_sta	**Indicates ddr_reserve_mode is successful** 0: Failed 1: Succeed
10	rg_dramc_conf_iso	**ddr_reserve_mode related register**
9	rg_dramc_iso	**ddr_reserve_mode related register**
8	rg_dramc_sref	**ddr_reserve_mode related register**
7:4	rg_dramc_timeout	**ddr_reserve_mode related register**
1	rg_mcu_lath_en	
0	rg_ddr_protect_en	**Enables ddr_protect_mode** When ddr_reserve_mode is enabled, ddr_protect_en will be over-ridden automatically. 0: Disable 1: Enable

10007050 WDT_INTERCORE_SYNC Intercore Sync Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rg_intercore_sync[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	rg_intercore_sync[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	rg_intercore_sync	Register for intercore sync usage

10007054 WDT_INTERCORE_SYNC_SET Set Control of Intercore Sync Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rg_intercore_sync_set[31:16]															
Type	Other															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	rg_intercore_sync_set[15:0]															

Type	Other															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	rg_intercore_sync_set	Sets 1'b1 to the correspondent bit of rg_intercore_sync.

10007058 WDT_INTERCORE_SYNC_CLR Clear Control of Intercore Sync Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	rg_intercore_sync_clr[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	rg_intercore_sync_clr[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	rg_intercore_sync_clr	Set 1'b0 to the correspondent bit of rg_intercore_sync

10000310 RESET_DEGLITCH_KEY Reset Deglitch Enable Key Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	dgrst_en_key[31:16]															
Type	RW															
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	dgrst_en_key[15:0]															
Type	RW															
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit(s)	Name	Description
31:0	dgrst_en_key	Write 0x67D2_A357 to enable reset deglitch.

3.5 Programming Guide

3.5.1 TOPRGU Initial

Enable dual mode reset when TOPRGU is first initialized. Because WDT_MODE will not be reset and the dual mode will be disabled if system reset is triggered through WDT_SWRST, these registers will only be reset by SYSRSTB.

The following registers will not be reset by TOPRGU.

- WDT_MODE
- WDT_STA
- WDT_NONRST_REG

- WDT_NONRST_REG2
- WDT_REQ_MODE
- WDT_REQ_IRQ_EN
- WDT_DEBUG_CTL

3.5.2 Watchdog Timer

- Trigger WDT_RESTART right after WDT_LENGTH is updated.
- WDT_SWRST can be triggered without wdt_en set to 1'b1.
- It is recommended to trigger WDT_RESTART before setting wdt_en to 1'b1.

3.5.3 IRQ Mode

Dual mode reset is default on. Therefore, all reset requests are default with IRQ mode enabled. This means the interrupt is triggered instead of triggering system reset immediately. If you would like to trigger system reset instead of interrupt, change the corresponding configuration of each reset request.

Each reset request can be configured as reset or IRQ separately.

3.5.4 CONNSYS Watchdog Timeout

CONNSYS has its own watchdog timer. When the watchdog timers expire, it notify AP through interrupts. AP then asserts software reset to CONNSYS.

- CONSYS
  1. conn_rst = 1'b1
  2. Wait for 2T 32kHz then set md_rst/conn_rst = 1'b0.

3.5.5 Dual Mode Reset

Dual mode reset is system reset after TOPRGU triggers interrupt. The watchdog timer needs to be enabled to complete this function.

In this mode, the watchdog timer will be ***AUTO-RESTART*** after interrupt is triggered. AP needs to clear WDT_STA after receiving interrupt from TOPRGU, or system reset will be triggered after watchdog timer expires.

1. Set wdt_en = 1'b1.
2. Set dual_mode = 1'b1.
3. Set wdt_irq, thermal_irq, scpsys_irq or debug_irq to 1'b1.

3.5.6 DDR Protect

DDR protect (rg_ddr_protect_en) is useless when DDR reserved mode is enabled.

3.5.7 DDR Reserved Mode Reset

DDR reserved mode keeps data in DDR during system reset. In order to complete this function, DRAMC, DRMC_CONF, DDRPHY_CONF and EMI_CONF (optional) will not be reset.

1. Enable DDR reserved mode when initializing TOPRGU.
2. Wait for system reset to be triggered.
3. [Optional] Check DDR reserved mode status (ddr_reserve_sta).
4. After system reset, release DRAMC_CONF protect (set rg_dramc_conf_iso = 1’b0).
5. Ensure related clocks of EMI, DRAMC, DDRPHY are ready (including PLL).
6. Wait for dramc_sref_sta = 1’b1.
7. Release DRAMC protect (set rg_dramc_iso = 1’b0).
8. Release DRAMC self-refresh control (set rg_dramc_sref = 1’b0).
9. Wait for dramc_sref_sta = 1’b0.

4 Peripheral Controller

4.1 Introduction

The peripheral controller is used to control the reset, clock and bus setting of peripheral sub-system. Each module inside the peripheral sub-system has its own software reset and clock gated control (power-down control).

4.2 Feature list

- Supports software reset control of each module inside peripheral sub-system
- Supports clock gated control of the modules insider peripheral sub-system by AP MCU
- Supports DCM control of peripheral sub-system
- Supports bus setting (bandwidth limit/way enable/…) of peripheral sub-system

4.3 Block Diagram

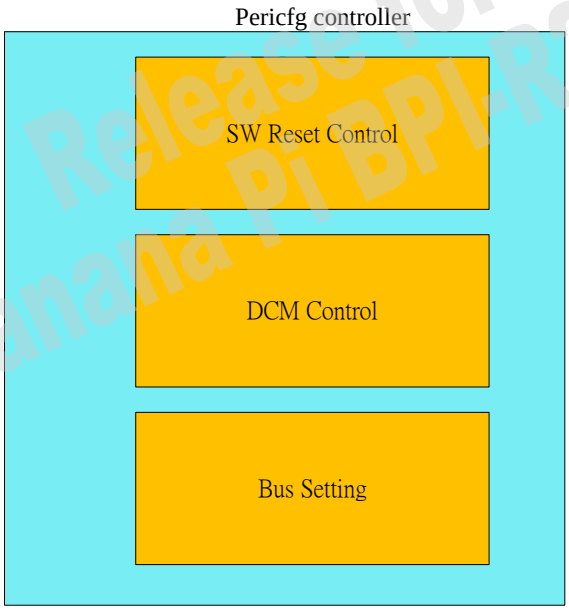


Figure 4-1: Block diagram of Pericfg Controller

4.4 Register Definition

Module name: PERICFG Base address: (+10003000h)

Address	Name	Width	Register Function
10003000	**PERI_GLOBAL**	32	**Peripheral Software Reset Registero**

	CON_RST0		
10003004	PERI_GLOBAL CON_RST1	32	Peripheral Software Reset Register1
10003008	PERI_GLOBAL CON_PDN0_SET	32	Peripheral Power Down0 Register Set
1000300C	PERI_GLOBAL CON_PDN1_SET	32	Peripheral Power Down1 Register Set
10003010	PERI_GLOBAL CON_PDN0_CLR	32	Peripheral Power Down0 Register Clear
10003014	PERI_GLOBAL CON_PDN1_CLR	32	Peripheral Power Down1 Register Clear
10003018	PERI_GLOBAL CON_PDN0_STA	32	Peripheral Power Down0 Register Status
1000301C	PERI_GLOBAL CON_PDN1_STA	32	Peripheral Power Down1 Register Status
10003020	PERI_GLOBAL CON_PDN_MD1_SET	32	Peripheral MD1 Power Down0 Register Set
10003024	PERI_GLOBAL CON_PDN_MD2_SET	32	Peripheral MD2 Power Down0 Register Set
10003028	PERI_GLOBAL CON_PDN_MD1_CLR	32	Peripheral MD1 Power Down0 Register Clear
1000302C	PERI_GLOBAL CON_PDN_MD2_CLR	32	Peripheral MD2 Power Down0 Register Clear
10003030	PERI_GLOBAL CON_PDN_MD1_STA	32	Peripheral MD1 Power Down0 Register Status
10003034	PERI_GLOBAL CON_PDN_MD2_STA	32	Peripheral MD2 Power Down0 Register Status
10003038	PERI_GLOBAL CON_PDN_MD_MASK	32	Peripheral MD Power Down1 Register MASK
10003050	PERI_GLOBAL CON_DCMCTL	32	Peripheral DCM Control Register
10003054	PERI_GLOBAL CON_DCMDBC	32	Peripheral DCM Debounce Counter
10003058	PERI_GLOBAL CON_DCMFSEL	32	Peripheral DCM Frequency Selection
1000305C	PERI_GLOBAL CON_CKSEL	32	Peripheral Clock Selection
10003200	PERIAXI_BUS_CTL1	32	Peripheral AXI Bus Control 1
10003204	PERIAXI_BUS_CTL2	32	Peripheral AXI BUS Control 2
10003208	PERIAXI_SI0_CTL	32	Peripheral AXI SI0 Control
1000320C	PERIAXI_SI1_CTL	32	Peripheral AXI SI1 Control

10003210	PERIAXI_MI_STA	32	Peripheral AXI MI Status
10003300	PERIAXI_AHB_LMT_CON1	32	Peripheral AHB Bus 0 Bandwidth Limiter
10003304	PERIAXI_AHB_LMT_CON2	32	Peripheral AHB Bus 0 Bandwidth Limiter
10003308	PERIAXI_AHB_LMT_CON3	32	Peripheral AHB Bus 1 Bandwidth Limiter
1000330C	PERIAXI_AHB_LMT_CON4	32	Peripheral AHB Bus 1 Bandwidth Limiter
10003310	PERIAXI_AHB_LMT_CON5	32	Peripheral AHB Bus 2 Bandwidth Limiter
10003314	PERIAXI_AHB_LMT_CON6	32	Peripheral AHB Bus 2 Bandwidth Limiter
10003318	PERIAXI_AHB_LMT_CON7	32	Peripheral AHB Bus 3 Bandwidth Limiter
1000331C	PERIAXI_AHB_LMT_CON8	32	Peripheral AHB Bus 3 Bandwidth Limiter
10003320	PERIAXI_AXI_LMT_CON1	32	Peripheral AXI MST0 Bandwidth Limiter
10003324	PERIAXI_AXI_LMT_CON2	32	Peripheral AXI MST1 Bandwidth Limiter
10003328	PERIAXI_AXI_LMT_CON3	32	Peripheral AXI MST2 Bandwidth Limiter
1000332C	PERIAXI_AXI_LMT_CON4	32	Peripheral AXI DMA Bandwidth Limiter
10003330	PERIAXI_AXI_LMT_CON5	32	Peripheral AXI DMA Bandwidth Limiter
10003334	PERIAXI_AXI_LMT_CON6	32	Peripheral AXI DMA Bandwidth Limiter
10003338	PERIAXI_AXI_LMT_CON7	32	Peripheral AXI DMA Bandwidth Limiter
1000333C	PERIAXI_AXI_LMT_CON8	32	Peripheral AXI DMA Bandwidth Limiter
10003400	PERI_USB_WAKEUP_DEC_CON0	32	Peripheral USB WAKEUP CONTROL0
1000340C	PERI_UART_CK_SOURCE_SEL	32	Peripheral UART CLOCK SOURCE SELECTION
10003420	PERI_ETH_NIC_CON	32	Peripheral ETH_NIC CONTROL
10003424	PERI_NFI_CK_SOURCE_SEL	32	Peripheral NFI CLOCK SOURCE SELECTION
10003428	PERI_NFI_MAC_CTRL	32	Peripheral NFI MAC CONTROL

10003000 **PERI_GLOBALCON_RST0** **Peripheral Software Reset Register0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ETH_SW	USB_SW			I2C3_SW	I2C2_SW	I2C1_SW_	I2C0_SW		MSDC1_S	MSDC0_		MSDC2_	THERM_

			_RST	_RST			_RST	_RST	RST	_RST		W_RST	SW_RST		SW_RST	SW_RST
Type			RW	RW			RW	RW	RW	RW		RW	RW		RW	RW
Reset			0	0			0	0	0	0		0	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NLI_SW_RST	NFI_SW_RST			DMA_SW_RST	AUXADC_SW_RST		PWM_SW_RST		BTIF_SW_RST	GCPU_SW_RST		UART3_SW_RST	UART2_SW_RST	UART1_SW_RST	UART0_SW_RST
Type	RW	RW			RW	RW		RW		RW	RW		RW	RW	RW	RW
Reset	0	0			0	0		0		0	0		0	0	0	0

Bit(s)	Mnemonic	Name	Description
29	**ETH_SW_RST**	ETH_SW_RST	**ETH software reset** 0: Not reset ETH 1: Reset ETH
28	**USB_SW_RST**	USB_SW_RST	**USB software reset** 0: Not reset USB 1: Reset USB
25	**I2C3_SW_RST**	I2C3_SW_RST	**I2C3 software reset** 0: Not reset I2C3 1: Reset I2C3
24	**I2C2_SW_RST**	I2C2_SW_RST	**I2C2 software reset** 0: Not reset I2C2 1: Reset I2C2
23	**I2C1_SW_RST**	I2C1_SW_RST	**I2C1 software reset** 0: Not reset I2C1 1: Reset I2C1
22	**I2C0_SW_RST**	I2C0_SW_RST	**I2C0 software reset** 0: Not reset I2C0 1: Reset I2C0
20	**MSDC1_SW_RST**	MSDC1_SW_RST	**MSDC1 software reset** 0: Not reset MSDC1 1: Reset MSDC1
19	**MSDC0_SW_RST**	MSDC0_SW_RST	**MSDC0 software reset** 0: Not reset MSDC0 1: Reset MSDC0
17	**MSDC2_SW_RST**	MSDC2_SW_RST	**MSDC2 software reset** 0: Not reset MSDC2 1: Reset MSDC2
16	**THERM_SW_RST**	THERM_SW_RST	**THERM software reset** 0: Not reset THERM 1: Reset THERM
15	**NLI_SW_RST**	NLI_SW_RST	**NLI software reset** 0: Not reset NLI 1: Reset NLI
14	**NFI_SW_RST**	NFI_SW_RST	**NFI software reset** 0: Not reset NFI 1: Reset NFI
11	**DMA_SW_RST**	DMA_SW_RST	**DMA software reset** 0: Not reset DMA

Bit(s)	Mnemonic	Name	Description
			1: Reset DMA
10	**AUXADC_SW_RST**	AUXADC_SW_RST	**AUXADC software reset** 0: Not reset AUXADC 1: Reset AUXADC
8	**PWM_SW_RST**	PWM_SW_RST	**PWM software reset** 0: Not reset PWM 1: Reset PWM
6	**BTIF_SW_RST**	BTIF_SW_RST	**BTIF software reset** 0: Not reset BTIF 1: Reset BTIF
5	**GCPU_SW_RST**	GCPU_SW_RST	**GCPU software reset** 0: Not reset GCPU 1: Reset GCPU
3	**UART3_SW_RST**	UART3_SW_RST	**UART3 software reset** 0: Not reset UART3 1: Reset UART3
2	**UART2_SW_RST**	UART2_SW_RST	**UART2 software reset** 0: Not reset UART2 1: Reset UART2
1	**UART1_SW_RST**	UART1_SW_RST	**UART1 software reset** 0: Not reset UART1 1: Reset UART1
0	**UART0_SW_RST**	UART0_SW_RST	**UART0 software reset** 0: Not reset UART0 1: Reset UART0

10003004 **PERI_GLOBALCON_RST1** **Peripheral Software Reset Register1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															**SPIO_SW_RST**	
Type															RW	
Reset															0	

Bit(s)	Mnemonic	Name	Description
1	**SPIO_SW_RST**	SPIO_SW_RST	**SPIO software reset** 0: Not reset SPIO 1: Reset SPIO

10003008	PERI_GLOBALCON_PDNo_SET	Peripheral Power Downo Register Set	00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USBo_MCU_PDN	ETH_PDN	SPIo_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2Co_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UARTo_PDN	NLI_PDN
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	o
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USBo_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o

Bit(s)	Mnemonic	Name	Description
31	USB_SLV_PDN	USB_SLV_PDN	**USB_SLV power down** o: keep original value 1: Power down USB_SLV
30	USB1_MCU_PDN	USB1_MCU_PDN	**USB1_MCU power down** o: keep original value 1: Power down USB1_MCU
29	USBo_MCU_PDN	USBo_MCU_PDN	**USBo_MCU power down** o: keep original value 1: Power down USBo_MCU
28	ETH_PDN	ETH_PDN	**ETH power down** o: keep original value 1: Power down ETH
27	SPIo_PDN	SPIo_PDN	**SPIo power down** o: keep original value 1: Power down SPIo
26	AUXADC_PDN	AUXADC_PDN	**AUXADC power down** o: keep original value 1: Power down AUXADC
25	I2C3_PDN	I2C3_PDN	**I2C3 power down** o: keep original value 1: Power down I2C3
24	I2C2_PDN	I2C2_PDN	**I2C2 power down** o: keep original value 1: Power down I2C2
23	I2C1_PDN	I2C1_PDN	**I2C1 power down** o: keep original value 1: Power down I2C1
22	I2Co_PDN	I2Co_PDN	**I2Co power down** o: keep original value 1: Power down I2Co
21	BTIF_PD	BTIF_PDN	**BTIF power down**

Bit(s)	Mnemonic	Name	Description
	N		0: keep original value 1: Power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 power down** 0: keep original value 1: Power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value 1: Power down PWM4
4	**PWM3_P**	PWM3_PDN	**PWM3 power down**

Bit(s)	Mnemonic	Name	Description
	DN		0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM power down** 0: keep original value 1: Power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**AUXADC power down** 0: keep original value 1: Power down NFI

1000300C **PERI_GLOBALCON_PDN1_SET** **Peripheral Power Down1 Register Set** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														**NFIPAD_PDN**	**NFI_ECC_PDN**	**GCPU_PDN**
Type														WO	WO	WO
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2	**NFIPAD_PDN**	NFIPAD_PDN	**NFIPAD power down** 0: keep original value 1: Power down NFIPAD
1	**NFI_ECC_PDN**	NFI_ECC_PDN	**NFI_ECC power down** 0: keep original value 1: Power down NFI_ECC
0	**GCPU_PDN**	GCPU_PDN	**GCPU power down** 0: keep original value 1: Power down GCPU

10003010 **PERI_GLOBALCON_PDNo_CLR** **Peripheral Power Downo Register Clear** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Nam	US	US	US	ET	SPI	AU	I2C	I2C	I2C	I2C	BTI	UA	UA	UA	UA	NLI

e	B_SLV_PDN	B1_MCU_PDN	Bo_MCU_PDN	H_PDN	o_PDN	XADC_PDN	3_PDN	2_PDN	1_PDN	o_PDN	F_PDN	RT3_PDN	RT2_PDN	RT1_PDN	RTo_PDN	_PDN
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USBo_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**USB_SLV_PDN**	USB_SLV_PDN	**USB_SLV clear power down** 0: keep original value 1: clear power down USB_SLV
30	**USB1_MCU_PDN**	USB1_MCU_PDN	**USB1_MCU clear power down** 0: keep original value 1: clear power down USB1_MCU
29	**USBo_MCU_PDN**	USBo_MCU_PDN	**USBo_MCU clear power down** 0: keep original value 1: clear power down USBo_MCU
28	**ETH_PDN**	ETH_PDN	**ETH clear power down** 0: keep original value 1: clear power down ETH
27	**SPIo_PDN**	SPIo_PDN	**SPIo clear power down** 0: keep original value 1: clear power down SPIo
26	**AUXADC_PDN**	AUXADC_PDN	**AUXADC clear power down** 0: keep original value 1: clear power down AUXADC
25	**I2C3_PDN**	I2C3_PDN	**I2C3 clear power down** 0: keep original value 1: clear power down I2C3
24	**I2C2_PDN**	I2C2_PDN	**I2C2 clear power down** 0: keep original value 1: clear power down I2C2
23	**I2C1_PDN**	I2C1_PDN	**I2C1 clear power down** 0: keep original value 1: clear power down I2C1
22	**I2Co_PDN**	I2Co_PDN	**I2Co clear power down** 0: keep original value 1: clear power down I2Co
21	**BTIF_PDN**	BTIF_PDN	**BTIF clear power down** 0: keep original value 1: clear power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 clear power down** 0: keep original value

Bit(s)	Mnemonic	Name	Description
			1: clear power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 clear power down** 0: keep original value 1: clear power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 clear power down** 0: keep original value 1: clear power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 clear power down** 0: keep original value 1: clear power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI clear power down** 0: keep original value 1: clear power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 clear power down** 0: keep original value 1: clear power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 clear power down** 0: keep original value 1: clear power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 clear power down** 0: keep original value 1: clear power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA clear power down** 0: keep original value 1: clear power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 clear power down** 0: keep original value 1: clear power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 clear power down** 0: keep original value 1: clear power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM clear power down** 0: keep original value 1: clear power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 clear power down** 0: keep original value 1: clear power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 clear power down** 0: keep original value 1: clear power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 clear power down** 0: keep original value 1: clear power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 clear power down** 0: keep original value 1: clear power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 clear power down** 0: keep original value 1: clear power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 clear power down** 0: keep original value

Bit(s)	Mnemonic	Name	Description
			1: clear power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 clear power down** 0: keep original value 1: clear power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM clear power down** 0: keep original value 1: clear power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI clear power down** 0: keep original value 1: clear power down NFI

10003014 **PERI_GLOBALCON_PDN1_CLR** **Peripheral Power Down1 Register Clear** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														**NFIPAD_PDN**	**NFI_ECC_PDN**	**GCPU_PDN**
Type														W1C	W1C	W1C
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2	**NFIPAD_PDN**	NFIPAD_PDN	**NFIPAD clear power down** 0: keep original value 1: clear power down NFIPAD
1	**NFI_ECC_PDN**	NFI_ECC_PDN	**NFI_ECC clear power down** 0: keep original value 1: clear power down NFI_ECC
0	**GCPU_PDN**	GCPU_PDN	**GCPU clear power down** 0: keep original value 1: clear power down GCPU

10003018 **PERI_GLOBALCON_PDNo_STA** **Peripheral Power Downo Register Status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**USB_SLV_PD**	**USB1_MCU_PD**	**USBo_MCU_P**	**ETH_PDN**	**SPIo_PDN**	**AUXADC_PDN**	**I2C3_PDN**	**I2C2_PDN**	**I2C1_PDN**	**I2Co_PDN**	**BTIF_PDN**	**UART3_PDN**	**UART2_PDN**	**UART1_PDN**	**UARTo_PDN**	**NLI_PDN**

	N	N	DN													
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USB0_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**USB_SLV_PDN**	USB_SLV_PDN	**USB_SLV power down** 0: keep original value 1: Power down USB_SLV
30	**USB1_MCU_PDN**	USB1_MCU_PDN	**USB1_MCU power down** 0: keep original value 1: Power down USB1_MCU
29	**USB0_MCU_PDN**	USB0_MCU_PDN	**USB0_MCU power down** 0: keep original value 1: Power down USB0_MCU
28	**ETH_PDN**	ETH_PDN	**ETH power down** 0: keep original value 1: Power down ETH
27	**SPI0_PDN**	SPI0_PDN	**SPI0 power down** 0: keep original value 1: Power down SPI0
26	**AUXADC_PDN**	AUXADC_PDN	**AUXADC power down** 0: keep original value 1: Power down AUXADC
25	**I2C3_PDN**	I2C3_PDN	**I2C23 power down** 0: keep original value 1: Power down I2C3
24	**I2C2_PDN**	I2C2_PDN	**I2C2 power down** 0: keep original value 1: Power down I2C2
23	**I2C1_PDN**	I2C1_PDN	**I2C1 power down** 0: keep original value 1: Power down I2C1
22	**I2C0_PDN**	I2C0_PDN	**I2C0 power down** 0: keep original value 1: Power down I2C0
21	**BTIF_PDN**	BTIF_PDN	**BTIF power down** 0: keep original value 1: Power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	**UART1_**	UART1_PDN	**UART1 power down**

Bit(s)	Mnemonic	Name	Description
	PDN		0: keep original value 1: Power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value 1: Power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 power down** 0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_**	THERM_PDN	**THERM power down**

Bit(s)	Mnemonic	Name	Description
	PDN		0: keep original value 1: Power down THERM
0	NFI_PDN	NFI_PDN_SET	**NFI power down** 0: keep original value 1: Power down NFI

1000301C PERI_GLOBALCON_PDN1_STA Peripheral Power Down1 Register Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														NFIPAD_PDN	NFI_ECC_PDN	GCPU_PDN
Type														RO	RO	RO
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2	NFIPAD_PDN	NFIPAD_PDN	**NFIPAD power down** 0: keep original value 1: Power down NFIPAD
1	NFI_ECC_PDN	NFI_ECC_PDN	**NFI_ECC power down** 0: keep original value 1: Power down NFI_ECC
0	GCPU_PDN	GCPU_PDN	**GCPU power down** 0: keep original value 1: Power down GCPU

10003020 PERI_GLOBALCON_PDN_MD1_SET Peripheral MD1 Power Down0 Register Set 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USB0_MCU_PDN	ETH_PDN	SPI0_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2C0_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UART0_PDN	NLI_PDN
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30	MSDC20	MSDC20	AP_DMA	USB1_PD	USB0_P	PWM_PD	PWM7_P	PWM6_P	PWM5_P	PWM4_P	PWM3_P	PWM2_P	PWM1_P	THERM_	NFI_PDN

	_1_PDN	_2_PDN	_1_PDN	_PDN	N	DN	N	DN	DN	DN	DN	DN	DN	DN	PDN	
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	USB_SLV_PDN	USB_SLV_PDN	**USB_SLV power down** 0: keep original value 1: Power down USB_SLV
30	USB1_MCU_PDN	USB1_MCU_PDN	**USB1_MCU power down** 0: keep original value 1: Power down USB1_MCU
29	USB0_MCU_PDN	USB0_MCU_PDN	**USB0_MCU power down** 0: keep original value 1: Power down USB0_MCU
28	ETH_PDN	ETH_PDN	**ETH power down** 0: keep original value 1: Power down ETH
27	SPI0_PDN	SPI0_PDN	**SPI0 power down** 0: keep original value 1: Power down SPI0
26	AUXADC_PDN	AUXADC_PDN	**AUXADC power down** 0: keep original value 1: Power down AUXADC
25	I2C3_PDN	I2C3_PDN	**I2C3 power down** 0: keep original value 1: Power down I2C3
24	I2C2_PDN	I2C2_PDN	**I2C2 power down** 0: keep original value 1: Power down I2C2
23	I2C1_PDN	I2C1_PDN	**I2C1 power down** 0: keep original value 1: Power down I2C1
22	I2C0_PDN	I2C0_PDN	**I2C0 power down** 0: keep original value 1: Power down I2C0
21	BTIF_PDN	BTIF_PDN	**BTIF power down** 0: keep original value 1: Power down BTIF
20	UART3_PDN	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	UART2_PDN	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	UART1_PDN	UART1_PDN	**UART1 power down** 0: keep original value 1: Power down UART1
17	UART0_PDN	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0

Bit(s)	Mnemonic	Name	Description
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value 1: Power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 power down** 0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM power down** 0: keep original value 1: Power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI power down** 0: keep original value 1: Power down NFI

10003024 **PERI_GLOBALCON_PDN_MD2_SET** **Peripheral MD2 Power Downo Register Set** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USBo_MCU_PDN	ETH_PDN	SPIo_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2Co_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UART0_PDN	NLI_PDN
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USBo_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**USB_SLV_PDN**	USB_SLV_PDN	**USB_SLV power down** 0: keep original value 1: Power down USB_SLV
30	**USB1_MCU_PDN**	USB1_MCU_PDN	**USB1_MCU power down** 0: keep original value 1: Power down USB1_MCU
29	**USBo_MCU_PDN**	USBo_MCU_PDN	**USBo_MCU power down** 0: keep original value 1: Power down USBo_MCU
28	**ETH_PDN**	ETH_PDN	**ETH power down** 0: keep original value 1: Power down ETH
27	**SPIo_PDN**	SPIo_PDN	**SPIo power down** 0: keep original value 1: Power down SPIo
26	**AUXADC_PDN**	AUXADC_PDN	**AUXADC power down** 0: keep original value 1: Power down AUXADC
25	**I2C3_PDN**	I2C3_PDN	**I2C3 power down** 0: keep original value 1: Power down I2C3
24	**I2C2_PDN**	I2C2_PDN	**I2C2 power down** 0: keep original value 1: Power down I2C2
23	**I2C1_PDN**	I2C1_PDN	**I2C1 power down** 0: keep original value 1: Power down I2C1
22	**I2Co_PDN**	I2Co_PDN	**I2Co power down** 0: keep original value

Bit(s)	Mnemonic	Name	Description
			1: Power down I2C0
21	**BTIF_PDN**	BTIF_PDN	**BTIF power down** 0: keep original value 1: Power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 power down** 0: keep original value 1: Power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value

Bit(s)	Mnemonic	Name	Description
			1: Power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 power down** 0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM power down** 0: keep original value 1: Power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI power down** 0: keep original value 1: Power down NFI

10003028 **PERI_GLOBALCON_PDN_MD1_CLR** **Peripheral MD1 Power Down0 Register Clear** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**USB_SLV_PDN**	**USB1_MCU_PDN**	**USB0_MCU_PDN**	**ETH_PDN**	**SPI0_PDN**	**AUXADC_PDN**	**I2C3_PDN**	**I2C2_PDN**	**I2C1_PDN**	**I2C0_PDN**	**BTIF_PDN**	**UART3_PDN**	**UART2_PDN**	**UART1_PDN**	**UART0_PDN**	**NLI_PDN**
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**MSDC30_1_PDN**	**MSDC20_2_PDN**	**MSDC20_1_PDN**	**AP_DMA_PDN**	**USB1_PDN**	**USB0_PDN**	**PWM_PDN**	**PWM7_PDN**	**PWM6_PDN**	**PWM5_PDN**	**PWM4_PDN**	**PWM3_PDN**	**PWM2_PDN**	**PWM1_PDN**	**THERM_PDN**	**NFI_PDN**
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**USB_SLV_PDN**	USB_SLV_PDN	**USB_SLV clear power down** 0: keep original value 1: clear power down USB_SLV
30	**USB1_MCU_PDN**	USB1_MCU_PDN	**USB1_MCU clear power down** 0: keep original value 1: clear power down USB1_MCU
29	**USB0_MCU_PDN**	USB0_MCU_PDN	**USB0_MCU clear power down** 0: keep original value 1: clear power down USB0_MCU

Bit(s)	Mnemonic	Name	Description
28	**ETH_PDN**	ETH_PDN	**ETH clear power down** 0: keep original value 1: clear power down ETH
27	**SPI0_PDN**	SPI0_PDN	**SPI0 clear power down** 0: keep original value 1: clear power down SPI0
26	**AUXADC_PDN**	AUXADC_PDN	**AUXADC clear power down** 0: keep original value 1: clear power down AUXADC
25	**I2C3_PDN**	I2C3_PDN	**I2C3 clear power down** 0: keep original value 1: clear power down I2C3
24	**I2C2_PDN**	I2C2_PDN	**I2C2 clear power down** 0: keep original value 1: clear power down I2C2
23	**I2C1_PDN**	I2C1_PDN	**I2C1 clear power down** 0: keep original value 1: clear power down I2C1
22	**I2C0_PDN**	I2C0_PDN	**I2C0 clear power down** 0: keep original value 1: clear power down I2C0
21	**BTIF_PDN**	BTIF_PDN	**BTIF clear power down** 0: keep original value 1: clear power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 clear power down** 0: keep original value 1: clear power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 clear power down** 0: keep original value 1: clear power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 clear power down** 0: keep original value 1: clear power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 clear power down** 0: keep original value 1: clear power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI clear power down** 0: keep original value 1: clear power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 clear power down** 0: keep original value 1: clear power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 clear power down** 0: keep original value 1: clear power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 clear power down** 0: keep original value 1: clear power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA clear power down** 0: keep original value 1: clear power down AP_DMA

Bit(s)	Mnemonic	Name	Description
11	**USB1_PDN**	USB1_PDN	**USB1 clear power down** 0: keep original value 1: clear power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 clear power down** 0: keep original value 1: clear power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM clear power down** 0: keep original value 1: clear power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 clear power down** 0: keep original value 1: clear power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 clear power down** 0: keep original value 1: clear power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 clear power down** 0: keep original value 1: clear power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 clear power down** 0: keep original value 1: clear power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 clear power down** 0: keep original value 1: clear power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 clear power down** 0: keep original value 1: clear power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 clear power down** 0: keep original value 1: clear power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM clear power down** 0: keep original value 1: clear power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI clear power down** 0: keep original value 1: clear power down NFI

1000302C **PERI_GLOBALCON_PDN_MD2_CLR** **Peripheral MD2 Power Down0 Register Clear** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USB0_MCU_PDN	ETH_PDN	SPI0_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2C0_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UART0_PDN	NLI_PDN
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USB0_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	USB_SLV_PDN	USB_SLV_PDN	**USB_SLV clear power down** 0: keep original value 1: clear power down USB_SLV
30	USB1_MCU_PDN	USB1_MCU_PDN	**USB1_MCU clear power down** 0: keep original value 1: clear power down USB1_MCU
29	USB0_MCU_PDN	USB0_MCU_PDN	**USB0_MCU clear power down** 0: keep original value 1: clear power down USB0_MCU
28	ETH_PDN	ETH_PDN	**ETH clear power down** 0: keep original value 1: clear power down ETH
27	SPI0_PDN	SPI0_PDN	**SPI0 clear power down** 0: keep original value 1: clear power down SPI0
26	AUXADC_PDN	AUXADC_PDN	**AUXADC clear power down** 0: keep original value 1: clear power down AUXADC
25	I2C3_PDN	I2C3_PDN	**I2C3 clear power down** 0: keep original value 1: clear power down I2C3
24	I2C2_PDN	I2C2_PDN	**I2C2 clear power down** 0: keep original value 1: clear power down I2C2
23	I2C1_PDN	I2C1_PDN	**I2C1 clear power down** 0: keep original value 1: clear power down I2C1
22	I2C0_PDN	I2C0_PDN	**I2C0 clear power down** 0: keep original value 1: clear power down I2C0
21	BTIF_PDN	BTIF_PDN	**BTIF clear power down** 0: keep original value 1: clear power down BTIF
20	UART3_PDN	UART3_PDN	**UART3 clear power down** 0: keep original value 1: clear power down UART3
19	UART2_PDN	UART2_PDN	**UART2 clear power down** 0: keep original value 1: clear power down UART2
18	UART1_PDN	UART1_PDN	**UART1 clear power down** 0: keep original value 1: clear power down UART1

Bit(s)	Mnemonic	Name	Description
17	**UART0_PDN**	UART0_PDN	**UART0 clear power down** 0: keep original value 1: clear power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI clear power down** 0: keep original value 1: clear power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 clear power down** 0: keep original value 1: clear power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 clear power down** 0: keep original value 1: clear power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 clear power down** 0: keep original value 1: clear power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA clear power down** 0: keep original value 1: clear power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 clear power down** 0: keep original value 1: clear power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 clear power down** 0: keep original value 1: clear power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM clear power down** 0: keep original value 1: clear power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 clear power down** 0: keep original value 1: clear power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 clear power down** 0: keep original value 1: clear power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 clear power down** 0: keep original value 1: clear power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 clear power down** 0: keep original value 1: clear power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 clear power down** 0: keep original value 1: clear power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 clear power down** 0: keep original value 1: clear power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 clear power down** 0: keep original value 1: clear power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM clear power down** 0: keep original value 1: clear power down THERM

Bit(s)	Mnemonic	Name	Description
0	NFI_PDN	NFI_PDN_SET	**AUXADC clear power down** 0: keep original value 1: clear power down NFI

10003030 **PERI_GLOBALCON_PDN_MD1_STA** **Peripheral MD1 Power Down0 Register Status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USB0_MCU_PDN	ETH_PDN	SPI0_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2C0_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UART0_PDN	NLI_PDN
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USB0_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	USB_SLV_PDN	USB_SLV_PDN	**USB_SLV power down** 0: keep original value 1: Power down USB_SLV
30	USB1_MCU_PDN	USB1_MCU_PDN	**USB1_MCU power down** 0: keep original value 1: Power down USB1_MCU
29	USB0_MCU_PDN	USB0_MCU_PDN	**USB0_MCU power down** 0: keep original value 1: Power down USB0_MCU
28	ETH_PDN	ETH_PDN	**ETH power down** 0: keep original value 1: Power down ETH
27	SPI0_PDN	SPI0_PDN	**SPI0 power down** 0: keep original value 1: Power down SPI0
26	AUXADC_PDN	AUXADC_PDN	**AUXADC power down** 0: keep original value 1: Power down AUXADC
25	I2C3_PDN	I2C3_PDN	**I2C3 power down** 0: keep original value 1: Power down I2C3
24	I2C2_PDN	I2C2_PDN	**I2C2 power down** 0: keep original value 1: Power down I2C2

Bit(s)	Mnemonic	Name	Description
23	**I2C1_PDN**	I2C1_PDN	**I2C1 power down** 0: keep original value 1: Power down I2C1
22	**I2C0_PDN**	I2C0_PDN	**I2C0 power down** 0: keep original value 1: Power down I2C0
21	**BTIF_PDN**	BTIF_PDN	**BTIF power down** 0: keep original value 1: Power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 power down** 0: keep original value 1: Power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6

Bit(s)	Mnemonic	Name	Description
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value 1: Power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 power down** 0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM power down** 0: keep original value 1: Power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI power down** 0: keep original value 1: Power down NFI

10003034 **PERI_GLOBALCON_PDN_MD2_STA** **Peripheral MD2 Power Downo Register Status** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB_SLV_PDN	USB1_MCU_PDN	USBo_MCU_PDN	ETH_PDN	SPIo_PDN	AUXADC_PDN	I2C3_PDN	I2C2_PDN	I2C1_PDN	I2C0_PDN	BTIF_PDN	UART3_PDN	UART2_PDN	UART1_PDN	UARTo_PDN	NLI_PDN
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC30_1_PDN	MSDC20_2_PDN	MSDC20_1_PDN	AP_DMA_PDN	USB1_PDN	USBo_PDN	PWM_PDN	PWM7_PDN	PWM6_PDN	PWM5_PDN	PWM4_PDN	PWM3_PDN	PWM2_PDN	PWM1_PDN	THERM_PDN	NFI_PDN
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**USB_SLV_PDN**	USB_SLV_PDN	**USB_SLV power down** 0: keep original value 1: Power down USB_SLV
30	**USB1_MCU_PDN**	USB1_MCU_PDN	**USB1_MCU power down** 0: keep original value 1: Power down USB1_MCU

Bit(s)	Mnemonic	Name	Description
29	**USB0_MCU_PDN**	USB0_MCU_PDN	**USB0_MCU power down** 0: keep original value 1: Power down USB0_MCU
28	**ETH_PDN**	ETH_PDN	**ETH power down** 0: keep original value 1: Power down ETH
27	**SPI0_PDN**	SPI0_PDN	**SPI0 power down** 0: keep original value 1: Power down SPI0
26	**AUXADC_PDN**	AUXADC_PDN	**AUXADC power down** 0: keep original value 1: Power down AUXADC
25	**I2C3_PDN**	I2C3_PDN	**I2C3 power down** 0: keep original value 1: Power down I2C3
24	**I2C2_PDN**	I2C2_PDN	**I2C2 power down** 0: keep original value 1: Power down I2C2
23	**I2C1_PDN**	I2C1_PDN	**I2C1 power down** 0: keep original value 1: Power down I2C1
22	**I2C0_PDN**	I2C0_PDN	**I2C0 power down** 0: keep original value 1: Power down I2C0
21	**BTIF_PDN**	BTIF_PDN	**BTIF power down** 0: keep original value 1: Power down BTIF
20	**UART3_PDN**	UART3_PDN	**UART3 power down** 0: keep original value 1: Power down UART3
19	**UART2_PDN**	UART2_PDN	**UART2 power down** 0: keep original value 1: Power down UART2
18	**UART1_PDN**	UART1_PDN	**UART1 power down** 0: keep original value 1: Power down UART1
17	**UART0_PDN**	UART0_PDN	**UART0 power down** 0: keep original value 1: Power down UART0
16	**NLI_PDN**	NLI_PDN	**NLI power down** 0: keep original value 1: Power down NLI
15	**MSDC30_1_PDN**	MSDC30_2_PDN	**MSDC30_2 power down** 0: keep original value 1: Power down MSDC30_2
14	**MSDC20_2_PDN**	MSDC30_1_PDN	**MSDC30_1 power down** 0: keep original value 1: Power down MSDC30_1
13	**MSDC20_1_PDN**	MSDC30_0_PDN	**MSDC30_0 power down** 0: keep original value 1: Power down MSDC30_0

Bit(s)	Mnemonic	Name	Description
12	**AP_DMA_PDN**	AP_DMA_PDN	**AP_DMA power down** 0: keep original value 1: Power down AP_DMA
11	**USB1_PDN**	USB1_PDN	**USB1 power down** 0: keep original value 1: Power down USB1
10	**USB0_PDN**	USB0_PDN	**USB0 power down** 0: keep original value 1: Power down USB0
9	**PWM_PDN**	PWM_PDN	**PWM power down** 0: keep original value 1: Power down PWM7
8	**PWM7_PDN**	PWM7_PDN	**PWM7 power down** 0: keep original value 1: Power down PWM7
7	**PWM6_PDN**	PWM6_PDN	**PWM6 power down** 0: keep original value 1: Power down PWM6
6	**PWM5_PDN**	PWM5_PDN	**PWM5 power down** 0: keep original value 1: Power down PWM5
5	**PWM4_PDN**	PWM4_PDN	**PWM4 power down** 0: keep original value 1: Power down PWM4
4	**PWM3_PDN**	PWM3_PDN	**PWM3 power down** 0: keep original value 1: Power down PWM3
3	**PWM2_PDN**	PWM2_PDN	**PWM2 power down** 0: keep original value 1: Power down PWM2
2	**PWM1_PDN**	PWM1_PDN	**PWM1 power down** 0: keep original value 1: Power down PWM1
1	**THERM_PDN**	THERM_PDN	**THERM power down** 0: keep original value 1: Power down THERM
0	**NFI_PDN**	NFI_PDN_SET	**NFI power down** 0: keep original value 1: Power down NFI

10003038 **PERI_GLOBALCON_PDN_MD_MASK** **Peripheral MD Power Down1 Register MASK** **00000003**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															MD2_MASK	MD1_MASK
Type															RW	RW
Reset															1	1

Bit(s)	Mnemonic	Name	Description
1	MD2_MASK	MD2_MASK	**MD2 power down mask** 0: MD2 power down take effect 1: MD2 power down no effect
0	MD1_MASK	MD1_MASK	**MD1 power down mask** 0: MD1 power down take effect 1: MD1 power down no effect

10003050 **PERI_GLOBALCON_DCMCTL** **Peripheral DCM Control Register** **000000F2**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DCM_IDLE_BYPASS_EN					AHB_BUS_SLP_REQ						AXI_CLOCK_GATED_EN	DCM_ENABLE
Type				RW					RW						RW	RW
Reset				0	0	0	0	0	1	1	1	1			1	0

Bit(s)	Mnemonic	Name	Description
12:8	DCM_IDLE_BYPASS_EN	DCM_IDLE_BYPASS_EN	**DCM idle bypass enable** bit 0: 1 ap_dma idle bypass enable bit 1: 1 ap_hif idle bypass enable bit 2: 1 md_hif idle bypass enable bit 3: usb idle bypass enable bist 4: msdc idle bypass
7:4	AHB_BUS_SLP_REQ	AHB_BUS_SLP_REQ	**AHB bus DCM control** bit 0: 1 AHB0_BUS_ON bit 1: 1 AHB1_BUS_ON bit 2: 1 AHB2_BUS_ON bit 3: 1 AHB3_BUS_ON
1	AXI_CLOCK_GATED_EN	AXI_CLOCK_GATED_EN	**AXI clock gated** 0: Disable 1: Enable
0	DCM_EN	DCM_ENABLE	**DCM control setting**

Bit(s)	Mnemonic	Name	Description
	ABLE		0: Disable 1: Enable

10003054 PERI_GLOBALCON_DCMDBC Peripheral DCM Debounce Counter 000000FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DCM_DBC_ENABLE	DCM_DBC_CNT						
Type									RW	RW						
Reset									1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
7	DCM_DBC_ENABLE	DCM_DBC_ENABLE	**DCM debouncing control** 0: Disable 1: Enable
6:0	DCM_DBC_CNT	DCM_DBC_CNT	**DCM debouncing counter**

10003058 PERI_GLOBALCON_DCMFSEL Peripheral DCM Frequency Selection 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												DCM_FULL_FSEL				
Type												RW				
Reset												0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DCM_HALF_FSEL									DCM_QTR_FSEL		
Type					RW									RW		
Reset					0	0	0	0						0	0	0

Bit(s)	Mnemonic	Name	Description
20:16	DCM_FULL_FSEL	DCM_FULL_FSEL	**DCM frequency selection for full speed clock**
11:8	DCM_HALF_FSEL	DCM_HALF_FSEL	**DCM frequency selection for half speed clock**

Bit(s)	Mnemonic	Name	Description
2:0	**DCM_QTR_FSEL**	DCM_QTR_FSEL	**DCM frequency selection for quarter speed clock**

1000305C **PERI_GLOBALCON_CKSEL** **Peripheral Clock Selection** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**PERIBUS_CK_SEL**
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**PERIBUS_CK_SEL**	PERIBUS_CK_SEL	**PERIBUS clock** Suggest to set to "half clock" mode when eMMC4.5 HS200 mode is enable. This can enhance the periaxi/periahb data transfer bandwidth, thus the eMMC4.5 HS200 mode data bandwidth can be improved. 0: peribus operates at quarter clock 1: peribus operates at half clock

10003200 **PERIAXI_BUS_CTL1** **Peripheral AXI Bus Control 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**AHB0_SHARE_EN**				**AHB1_SHARE_EN**											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0								
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Mnemonic	Name	Description
31:28	**AHB0_SHARE_EN**	AHB0_SHARE_EN	**AHB buso share enable** Bit 0: PWM share enable Bit 1: SPM share enable Bit 2: NFI share enable

Bit(s)	Mnemonic	Name	Description
27:24	**AHB1_SHARE_EN**	AHB1_SHARE_EN	Bit 3: USB0 share enable **AHB bus1 share enable** Bit 1: DBGAHB share enable Bit 2: ETHERNET share enable Bit 3: USB1 share enable

10003204 **PERIAXI_BUS_CTL2** **Peripheral AXI BUS Control 2** **101FFFFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AHB2_SHARE_EN											AHB_SECURE_EN[6:2]				
Type	RW											RW				
Reset	0	0	0	1								1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AHB_SECURE_EN[1:0]		AHB_BUFFER_EN							AHB_MERGE_EN						
Type	RW		RW							RW						
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:28	**AHB2_SHARE_EN**	AHB2_SHARE_EN	**AHB bus2 share enable** Bit 0: THERM share enable Bit 1: MSDC1 share enable Bit 2: SPI1 share enable Bit 3: SPI0share enable
20:14	**AHB_SECURE_EN**	AHB_SECURE_EN	**AHB2AXI secure enable**
13:7	**AHB_BUFFER_EN**	AHB_BUFFER_EN	**AHB2AXI buffer enable**
6:0	**AHB_MERGE_EN**	AHB_MERGE_EN	**AHB2AXI merge enable**

10003208 **PERIAXI_SI0_CTL** **Peripheral AXI SI0 Control** **00000300**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PERIAXI_CG_DISABLE							PERIAXI_SIo_WR_OT_BUSY	PERIAXI_SIo_RD_OT_BUSY	PERIAXI_SIo_DFSLV_SET_RID_MI	PERIAXI_SIo_DFSLV_SET_BID_MI	PERIAXI_SIo_DFSLV_SET_WIRQ	PERIAXI_SIo_DFSLV_SET_RIRQ	PERIAXI_SIo_R_CHNL_SEL		

										SS	SS					
Type	RW							RO	RO	RO	RO	RO	RO	RO		
Reset	0							0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERIAXI_SIo_B_CHNL_SEL			PERIAXI_SIo_CTRL_BYPASS			PERIAXI_SIo_WAY_EN									PERIAXI_SIo_OUSTANDING_DISABLE
Type	RO			RW			RW									RW
Reset	0	0	0	0			1	1								0

Bit(s)	Mnemonic	Name	Description
31	**PERIAXI_CG_DISABLE**	PERIAXI_CG_DISABLE	**PERIAXI CG disable control** 0: Enable CG 1: Disable CG
24	**PERIAXI_SIo_WR_OT_BUSY**	PERIAXI_SIo_WR_OT_BUSY	**PERIAXI control bypass** 0: Disable 1: Enable
23	**PERIAXI_SIo_RD_OT_BUSY**	PERIAXI_SIo_RD_OT_BUSY	**PERIAXI control bypass** 0: Disable 1: Enable
22	**PERIAXI_SIo_DFSLV_SET_RID_MISS**	PERIAXI_SIo_DFSLV_SET_RID_MISS	**PERIAXI control bypass** 0: Disable 1: Enable
21	**PERIAXI_SIo_DFSLV_SET_BID_MISS**	PERIAXI_SIo_DFSLV_SET_BID_MISS	**PERIAXI control bypass** 0: Disable 1: Enable
20	**PERIAXI_SIo_DFSLV_SET_WIRQ**	PERIAXI_SIo_DFSLV_SET_WIRQ	**PERIAXI control bypass** 0: Disable 1: Enable
19	**PERIAXI_SIo_DFSLV_SET_RIRQ**	PERIAXI_SIo_DFSLV_SET_RIRQ	**PERIAXI control bypass** 0: Disable 1: Enable
18:16	**PERIAXI_SIo_R_CHNL_SEL**	PERIAXI_SIo_R_CHNL_SEL	**PERIAXI control bypass** 0: Disable 1: Enable
15:13	**PERIAXI_SIo_B_CHNL_SEL**	PERIAXI_SIo_B_CHNL_SEL	**PERIAXI control bypass** 0: Disable 1: Enable

Bit(s)	Mnemonic	Name	Description
12	**PERIAXI_SIo_CTRL_BYPASS**	PERIAXI_SIo_CTRL_BYPASS	**PERIAXI control bypass** 0: Disable 1: Enable
9:8	**PERIAXI_SIo_WAY_EN**	PERIAXI_SIo_WAY_EN	**PERIAXI SIo way enable**
0	**PERIAXI_SIo_OUSTANDING_DISABLE**	PERIAXI_SIo_OUSTANDING_DISABLE	**PERIAXI SIo oustanding disable**

1000320C **PERIAXI_SI1_CTL** **Peripheral AXI SI1 Control** **00000700**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								PERIAXI_SI1_WR_OT_BUSY	PERIAXI_SI1_RD_OT_BUSY	PERIAXI_SI1_DFSLV_SET_RID_MISS	PERIAXI_SI1_DFSLV_SET_BID_MISS	PERIAXI_SI1_DFSLV_SET_WIRQ	PERIAXI_SI1_DFSLV_SET_RIRQ	PERIAXI_SI1_RCHNL_SEL		
Type								RO	RO	RO	RO	RO	RO	RO		
Reset								0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PERIAXI_SI1_BCHNL_SEL			PERIAXI_SI1_CTRL_BYPASS		PERIAXI_SI1_WAY_EN										PERIAXI_SI1_OUSTANDING_DISABLE
Type	RO			RW		RW										RW
Reset	0	0	0	0		1	1	1								0

Bit(s)	Mnemonic	Name	Description
24	**PERIAXI_SI1_WR_OT_BUSY**	PERIAXI_SI1_WR_OT_BUSY	**PERIAXI control bypass** 0: Disable 1: Enable
23	**PERIAXI**	PERIAXI_SI1_R	**PERIAXI control bypass**

Bit(s)	Mnemonic	Name	Description
	_SI1_RD_OT_BUSY	D_OT_BUSY	0: Disable 1: Enable
22	**PERIAXI_SI1_DFSLV_SET_RID_MISS**	PERIAXI_SI1_DFSLV_SET_RID_MISS	**PERIAXI control bypass** 0: Disable 1: Enable
21	**PERIAXI_SI1_DFSLV_SET_BID_MISS**	PERIAXI_SI1_DFSLV_SET_BID_MISS	**PERIAXI control bypass** 0: Disable 1: Enable
20	**PERIAXI_SI1_DFSLV_SET_WIRQ**	PERIAXI_SI1_DFSLV_SET_WIRQ	**PERIAXI control bypass** 0: Disable 1: Enable
19	**PERIAXI_SI1_DFSLV_SET_RIRQ**	PERIAXI_SI1_DFSLV_SET_RIRQ	**PERIAXI control bypass** 0: Disable 1: Enable
18:16	**PERIAXI_SI1_R_CHNL_SEL**	PERIAXI_SI1_R_CHNL_SEL	**PERIAXI control bypass** 0: Disable 1: Enable
15:13	**PERIAXI_SI1_B_CHNL_SEL**	PERIAXI_SI1_B_CHNL_SEL	**PERIAXI control bypass** 0: Disable 1: Enable
12	**PERIAXI_SI1_CTRL_BYPASS**	PERIAXI_SI1_CTRL_BYPASS	**PERIAXI control bypass** 0: Disable 1: Enable
10:8	**PERIAXI_SI1_WAY_EN**	PERIAXI_SI1_WAY_EN	**PERIAXI SIo way enable**
0	**PERIAXI_SI1_OUSTANDING_DISABLE**	PERIAXI_SI1_OUSTANDING_DISABLE	**PERIAXI SIo oustanding disable**

10003210 PERIAXI_MI_STA Peripheral AXI MI Status 00000040

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**MI1_W_BUS**	**MI1_R_BUS**	**MI1_ERRMI**	**MI1_ERRMI**	**MIo_W_BU**	**MIo_R_BU**	**MIo_ERRM**	**MIo_ERRM**

									Y	Y	D_SET_BIRQ	D_SET_RIRQ	SY	SY	ID_SET_BIRQ	ID_SET_RIRQ
Type									RO	RO	RO	RO	RO	RO	RO	RO
Reset									0	1	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	MI1_W_BUSY	MI1_W_BUSY	**PERIAXI MI1 Write Busy** 0: MI1 not write busy 1: MI1 write busy
6	MI1_R_BUSY	MI1_R_BUSY	**PERIAXI MI1 Read Busy** 0: MI1 not read busy 1: MI1 read busy
5	MI1_ERRMID_SET_BIRQ	MI1_ERRMID_SET_BIRQ	**PERIAXI MI1 Write Error** 0: MI1 no wite error MID 1: MI1 write error MID
4	MI1_ERRMID_SET_RIRQ	MI1_ERRMID_SET_RIRQ	**PERIAXI MI1 Read Error** 0: MI1 no read error MID 1: MI1 read error MID
3	MI0_W_BUSY	MI0_W_BUSY	**PERIAXI MI0 Write Busy** 0: MI0 not write busy 1: MI0 write busy
2	MI0_R_BUSY	MI0_R_BUSY	**PERIAXI MI0 Read Busy** 0: MI0 not read busy 1: MI0 read busy
1	MI0_ERRMID_SET_BIRQ	MI0_ERRMID_SET_BIRQ	**PERIAXI MI0 Write Error** 0: MI0 no wite error MID 1: MI0 write error MID
0	MI0_ERRMID_SET_RIRQ	MI0_ERRMID_SET_RIRQ	**PERIAXI MI0 Read Error** 0: MI0 no read error MID 1: MI0 read error MID

10003300 **PERIAXI_AHB_LMT_CON1** **Peripheral AHB Bus 0 Bandwidth Limiter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						MSDC0_EMI_ULTRA	BUFFER_MODE	BUFFER_EN	SOFT_LIMIT_EN				FILTER_CKEN	FILTER_LEN		FILTER_EN
Type						RW	RW	RW	RW				RW	RW		RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
10	**MSDCo_EMI_ULTRA**	MSDCo_EMI_ULTRA	**msdco AHB master read/write cmd/data ultra attribute to EMI.** Suggest to enable when the eMMC4.5 HS200 mode is enable. After enable it, every MSDCo command will become ultra priority in the view of EMI arbitration. 0: disable msdco AHB master read/write cmd/data with ultra attribute to EMI 1: enable msdco AHB master read/write cmd/data with ultra attribute to EMI
9	**BUFFER_MODE**	BUFFER_MODE	**The condition of early response of per2conn_ahb_bridge_f2s depends on hprot[2] signal of AHB** 0: always 1: conditional
8	**BUFFER_EN**	BUFFER_EN	**To enable early response ability of per2conn_ahb_bridge_f2s** 0: disable 1: enable early response
7:4	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AHB bus 0 bandwidth limiter enable**
3	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 0 bandwidth limiter filter clock enable**
2:1	**FILTER_LEN**	FILTER_LEN	**AHB bus 0 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
0	**FILTER_EN**	FILTER_EN	**AHB bus 0 bandwidth limiter filterenable**

10003304 PERIAXI_AHB_LMT_CON2 Peripheral AHB Bus 0 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USBoP_BUS_GNT_CNT								Audio_BUS_GNT_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDCo_BUS_GNT_CNT								NFI_BUS_GNT_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**USBoP_BUS_GNT_CNT**	USBoP_BUS_GNT_CNT	**USB oP bus grant count**
23:16	**Audio_BUS_GNT**	Audio_BUS_GNT_CNT	**Audio bus grant count**

Bit(s)	Mnemonic	Name	Description
	_CNT		
15:8	**MSDCo_BUS_GNT_CNT**	MSDCo_BUS_GNT_CNT	**MSDCo bus grant count**
7:0	**NFI_BUS_GNT_CNT**	NFI_BUS_GNT_CNT	**NFI bus grant count**

10003308 PERIAXI_AHB_LMT_CON3 Peripheral AHB Bus 1 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SOFT_LIMIT_EN				FILTER_CKEN	FILTER_LEN		FILTER_EN
Type									RW				RW	RW		RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AHB bus 1 bandwidth limiter enable**
3	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 1 bandwidth limiter filter clock enable**
2:1	**FILTER_LEN**	FILTER_LEN	**AHB bus 1 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
0	**FILTER_EN**	FILTER_EN	**AHB bus 1 bandwidth limiter filterenable**

1000330C PERIAXI_AHB_LMT_CON4 Peripheral AHB Bus 1 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPIo_BUS_GNT_CNT								MSDC2_BUS_GNT_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MSDC1_BUS_GNT_CNT								PWM_BUS_GNT_CNT							
Type	RW								RW							

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**SPIo_BUS_GNT_CNT**	SPIo_BUS_GNT_CNT	**SPIo bus grant count**
23:16	**MSDC2_BUS_GNT_CNT**	MSDC2_BUS_GNT_CNT	**MSDC2 bus grant count**
15:8	**MSDC1_BUS_GNT_CNT**	MSDC1_BUS_GNT_CNT	**MSDC1 bus grant count**
7:0	**PWM_BUS_GNT_CNT**	PWM_BUS_GNT_CNT	**PWM bus grant count**

10003310 PERIAXI_AHB_LMT_CON5 Peripheral AHB Bus 2 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**SOFT_LIMIT_EN**				**FILTER_CKEN**	**FILTER_LEN**		**FILTER_EN**
Type									RW				RW	RW		RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AHB bus 2 bandwidth limiter enable**
3	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 2 bandwidth limiter filter clock enable**
2:1	**FILTER_LEN**	FILTER_LEN	**AHB bus 2 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
0	**FILTER_EN**	FILTER_EN	**AHB bus 2 bandwidth limiter filterenable**

10003314 PERIAXI_AHB_LMT_CON6 Peripheral AHB Bus 2 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PTP_BUS_GNT_CNT								DEBUGTOP_BUS_GNT_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FHCTL_SPM_BUS_GNT_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0								

Bit(s)	Mnemonic	Name	Description
31:24	**PTP_BUS_GNT_CNT**	PTP_BUS_GNT_CNT	**PTP bus grant count**
23:16	**DEBUGTOP_BUS_GNT_CNT**	DEBUGTOP_BUS_GNT_CNT	**DEBUGTOP bus grant count**
15:8	**FHCTL_SPM_BUS_GNT_CNT**	FHCTL_SPM_BUS_GNT_CNT	**FHCTL_SPM bus grant count**

10003318 PERIAXI_AHB_LMT_CON7 Peripheral AHB Bus 3 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SOFT_LIMIT_EN				FILTER_CKEN	FILTER_LEN		FILTER_EN
Type									RW				RW	RW		RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AHB bus 3 bandwidth limiter enable**
3	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 3 bandwidth limiter filter clock enable**
2:1	**FILTER_LEN**	FILTER_LEN	**AHB bus 3 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
0	**FILTER_EN**	FILTER_EN	**AHB bus 3 bandwidth limiter filterenable**

1000331C PERIAXI_AHB_LMT_CON8 Peripheral AHB Bus 3 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB1P_BUS_GNT_CNT								ETH_BUS_GNT_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Mnemonic	Name	Description
31:24	**USB1P_BUS_GNT_CNT**	USB1P_BUS_GNT_CNT	**PTP bus grant count**
23:16	**ETH_BUS_GNT_CNT**	ETH_BUS_GNT_CNT	**DEBUGTOP bus grant count**

10003320 PERIAXI_AXI_LMT_CON1 Peripheral AXI MSTo Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_MSTo_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MSTo soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 0 bandwidth limiter filter clock enable**

Bit(s)	Mnemonic	Name	Description
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST0 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST0 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_MST0_BUS_GNT_CNT**	AXI_MST0_BUS_GNT_CNT	**AXI MST0 bus grant count**

10003324 **PERIAXI_AXI_LMT_CON2** **Peripheral AXI MST1 Bandwidth Limiter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																**SOFT_LIMIT_EN**
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					**FILTER_CKEN**	**BW_FILTER_EN**	**FILTER_LEN**		**AXI_MST1_BUS_GNT_CNT**							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MST1 soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 1 bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST1 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST1 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_MST1_BUS_GNT_CN**	AXI_MST1_BUS_GNT_CNT	**AXI MST1 bus grant count**

Bit(s)	Mnemonic	Name	Description
	T		

10003328 PERIAXI_AXI_LMT_CON3 Peripheral AXI MST2 Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_MST2_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	SOFT_LIMIT_EN	SOFT_LIMIT_EN	**AXI MST2 soft bandwidth limiter enable** 0: Disable 1: Enable
11	FILTER_CKEN	FILTER_CKEN	**AHB bus 2 bandwidth limiter filter clock enable**
10	BW_FILTER_EN	BW_FILTER_EN	**AXI MST2 bandwidth limiter enable** 0: Disable 1: Enable
9:8	FILTER_LEN	FILTER_LEN	**AXI MST2 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	AXI_MST2_BUS_GNT_CNT	AXI_MST2_BUS_GNT_CNT	**AXI MST2 bus grant count**

1000332C PERIAXI_AXI_LMT_CON4 Peripheral AXI DMA Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LI

																MIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_DMA_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI DMA soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AXI DMA bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI DMA bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI DMA bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_DMA_BUS_GNT_CNT**	AXI_DMA_BUS_GNT_CNT	**AXI DMA bus grant count**

10003330 **PERIAXI_AXI_LMT_CON5** **Peripheral AXI DMA Bandwidth Limiter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_DMA_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MST3 soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 3 bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST3 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST3 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_DMA_BUS_GNT_CNT**	AXI_DMA_BUS_GNT_CNT	**AXI MST3 bus grant count**

10003334 PERIAXI_AXI_LMT_CON6 Peripheral AXI DMA Bandwidth Limiter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_DMA_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MST3 soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 3 bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST3 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST3 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512

Bit(s)	Mnemonic	Name	Description
			10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_DMA_BUS_GNT_CNT**	AXI_DMA_BUS_GNT_CNT	**AXI MST3 bus grant count**

10003338 **PERIAXI_AXI_LMT_CON7** **Peripheral AXI DMA Bandwidth Limiter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_DMA_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MST3 soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 3 bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST3 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST3 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_DMA_BUS_GNT_CNT**	AXI_DMA_BUS_GNT_CNT	**AXI MST3 bus grant count**

1000333C **PERIAXI_AXI_LMT_CO** **Peripheral AXI DMA Bandwidth Limiter** **00000000**

N8

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SOFT_LIMIT_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					FILTER_CKEN	BW_FILTER_EN	FILTER_LEN		AXI_DMA_BUS_GNT_CNT							
Type					RW	RW	RW		RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16	**SOFT_LIMIT_EN**	SOFT_LIMIT_EN	**AXI MST3 soft bandwidth limiter enable** 0: Disable 1: Enable
11	**FILTER_CKEN**	FILTER_CKEN	**AHB bus 3 bandwidth limiter filter clock enable**
10	**BW_FILTER_EN**	BW_FILTER_EN	**AXI MST3 bandwidth limiter enable** 0: Disable 1: Enable
9:8	**FILTER_LEN**	FILTER_LEN	**AXI MST3 bandwidth limiter filter length** 00: Fliter length = 256 01: Filter length = 512 10: Filter length = 1024 11: Filter length = 2048
7:0	**AXI_DMA_BUS_GNT_CNT**	AXI_DMA_BUS_GNT_CNT	**AXI MST3 bus grant count**

10003400 **PERI_USB_WAKEUP_DEC_CON0** **Peripheral USB WAKEUP CONTROL0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												USB0_CDDEBOUNCE				USB0_CDEN
Type												RW				RW
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:1	**USB0_CDDEBOUNCE**	USB0_CDDEBOUNCE	**USB0_CDDEBOUNCE** usb0 debounce clock number: (0~15) 0000: 0 0001: 1 0010: 2 1111: 15
0	**USB0_CDEN**	USB0_CDEN	**USB0_CDEN** usb0 clock debounce enable 0: Disable 1: Enable

10003400C **PERI_UART_CK_SOURCE_SEL** **Peripheral UART CLOCK SOURCE SELECTION** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													UART_CK_SEL			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**UART_CK_SEL**	UART_CK_SEL	**UART_CK_SEL** Clock source selection for UART0 through UART3 0: 26MHz clock 1: 52MHz clock

10003420 **PERI_ETH_NIC_CON** **Peripheral ETH_NIC CONTROL** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													NIC_MII_MODE			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**NIC_MII_MODE**	NIC_MII_MODE	**NIC_MII_MODE**

10003424 **PERI_NFI_CK_SOURCE_SEL** **Peripheral NFI CLOCK SOURCE SELECTION** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																NFI_CK_SEL
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**NFI_CK_SEL**	NFI_CK_SEL	**NFI_CK_SEL** Clock source selection for NFI 0: axi clock 1: nfi pad 1x clock

10003428 **PERI_NFI_MAC_CTRL** **Peripheral NFI MAC CONTROL** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												NFI_MAC_CTRL				
Type												RW				
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:0	**NFI_MAC_CTRL**	NFI_MAC_CTRL	**NFI_MAC_CTRL**

5 External Interrupt Controller

5.1 Introduction

The external interrupt controller (EINTC) processes all off-chip interrupt sources and forwards interrupt request signals MCU.

5.2 Feature list

EINTC supports external interrupt signals and performs the following processes to the interrupt signals coming from external sources:

- Polarity inversion
- Edge/level trigger selection
- De-bounce with a configurable 32kHz clock (optional)

According to the register configuration, the external interrupt source will be forwarded to the Cortex-A7 built-in interrupt controller with different IRQ signals, eint_irq or eint_direct_irq. EINTC generates wake up events to SPM controller.

5.3 Block Diagram

This is the block diagram of the external interrupt controller. Every function block is controlled by the corresponding control register define in next section

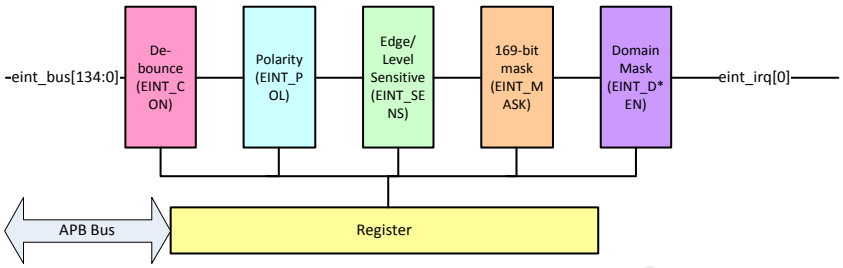


Figure 5-1: Block diagram of external interrupt controller

Normally the external interrupt source goes through the de-bounce unit which is driven by 32kHz clock and triggers the corresponding CPU with eint_irq. Therefore the minimum latency from eint_bus to eint_irq will be 30.52 µs. Since the latency introduced by the de-bounce module may be too long for

some applications, EINTC provides an alternative path which bypasses the de-bounce module and directly triggers the interrupt signals, eint_direct_irq[14:0], to AP MCU.

5.4 Register Definition

Module name: ap_cirq_eint_reg Base address: (+1000B000h)

Address	Name	Width	Register Function
1000B000	EINT_STA0	32	External interrupt status register
1000B004	EINT_STA1	32	External interrupt status register
1000B008	EINT_STA2	32	External interrupt status register
1000B00C	EINT_STA3	32	External interrupt status register
1000B010	EINT_STA4	32	External interrupt status register
1000B014	EINT_STA5	32	External interrupt status register
1000B040	EINT_ACK0	32	External interrupt acknowledge register
1000B044	EINT_ACK1	32	External interrupt acknowledge register
1000B048	EINT_ACK2	32	External interrupt acknowledge register
1000B04C	EINT_ACK3	32	External interrupt acknowledge register
1000B050	EINT_ACK4	32	External interrupt acknowledge register
1000B054	EINT_ACK5	32	External interrupt acknowledge register
1000B080	EINT_MASK0	32	External interrupt mask register
1000B084	EINT_MASK1	32	External interrupt mask register
1000B088	EINT_MASK2	32	External interrupt mask register
1000B08C	EINT_MASK3	32	External interrupt mask register
1000B090	EINT_MASK4	32	External interrupt mask register
1000B094	EINT_MASK5	32	External interrupt mask register
1000B0C0	EINT_MASK_SET0	32	External interrupt mask set register
1000B0C4	EINT_MASK_SET1	32	External interrupt mask set register
1000B0C8	EINT_MASK_SET2	32	External interrupt mask set register
1000B0CC	EINT_MASK_SET3	32	External interrupt mask set register
1000B0D0	EINT_MASK_SET4	32	External interrupt mask set register
1000B0D4	EINT_MASK_SET5	32	External interrupt mask set register
1000B100	EINT_MASK_CLR0	32	External interrupt mask set register
1000B104	EINT_MASK_CLR1	32	External interrupt mask set register
1000B108	EINT_MASK_CLR2	32	External interrupt mask set register
1000B10C	EINT_MASK_CLR3	32	External interrupt mask set register
1000B110	EINT_MASK_CLR4	32	External interrupt mask set register

Module name: ap_cirq_eint_reg Base address: (+1000B000h)

1000B114	EINT_MASK_CLR5	32	External interrupt mask set register
1000B140	EINT_SENS0	32	External interrupt sensitivity register
1000B144	EINT_SENS1	32	External interrupt sensitivity register
1000B148	EINT_SENS2	32	External interrupt sensitivity register
1000B14C	EINT_SENS3	32	External interrupt sensitivity register
1000B150	EINT_SENS4	32	External interrupt sensitivity register
1000B154	EINT_SENS5	32	External interrupt sensitivity register
1000B180	EINT_SENS_SET0	32	External interrupt sensitivity set register
1000B184	EINT_SENS_SET1	32	External interrupt sensitivity set register
1000B188	EINT_SENS_SET2	32	External interrupt sensitivity set register
1000B18C	EINT_SENS_SET3	32	External interrupt sensitivity set register
1000B190	EINT_SENS_SET4	32	External interrupt sensitivity set register
1000B194	EINT_SENS_SET5	32	External interrupt sensitivity set register
1000B1C0	EINT_SENS_CLR0	32	External interrupt sensitivity clear register
1000B1C4	EINT_SENS_CLR1	32	External interrupt sensitivity clear register
1000B1C8	EINT_SENS_CLR2	32	External interrupt sensitivity clear register
1000B1CC	EINT_SENS_CLR3	32	External interrupt sensitivity clear register
1000B1D0	EINT_SENS_CLR4	32	External interrupt sensitivity clear register
1000B1D4	EINT_SENS_CLR5	32	External interrupt sensitivity clear register
1000B200	EINT_SOFT0	32	Software interrupt register
1000B204	EINT_SOFT1	32	Software interrupt register
1000B208	EINT_SOFT2	32	Software interrupt register
1000B20C	EINT_SOFT3	32	Software interrupt register
1000B210	EINT_SOFT4	32	Software interrupt register
1000B214	EINT_SOFT5	32	Software interrupt register
1000B240	EINT_SOFT_SET0	32	Software interrupt set register
1000B244	EINT_SOFT_SET1	32	Software interrupt set register
1000B248	EINT_SOFT_SET2	32	Software interrupt set register
1000B24C	EINT_SOFT_SET3	32	Software interrupt set register
1000B250	EINT_SOFT_SET4	32	Software interrupt set register
1000B254	EINT_SOFT_SET5	32	Software interrupt set register
1000B280	EINT_SOFT_C	32	Software interrupt clear register

Module name: ap_cirq_eint_reg Base address: (+1000B000h)

	LR0		
1000B284	EINT_SOFT_CLR1	32	Software interrupt clear register
1000B288	EINT_SOFT_CLR2	32	Software interrupt clear register
1000B28C	EINT_SOFT_CLR3	32	Software interrupt clear register
1000B290	EINT_SOFT_CLR4	32	Software interrupt clear register
1000B294	EINT_SOFT_CLR5	32	Software interrupt clear register
1000B300	EINT_POL0	32	External interrupt polarity register
1000B304	EINT_POL1	32	External interrupt polarity register
1000B308	EINT_POL2	32	External interrupt polarity register
1000B30C	EINT_POL3	32	External interrupt polarity register
1000B310	EINT_POL4	32	External interrupt polarity register
1000B314	EINT_POL5	32	External interrupt polarity register
1000B340	EINT_POL_SET0	32	External interrupt polarity set register
1000B344	EINT_POL_SET1	32	External interrupt polarity set register
1000B348	EINT_POL_SET2	32	External interrupt polarity set register
1000B34C	EINT_POL_SET3	32	External interrupt polarity set register
1000B350	EINT_POL_SET4	32	External interrupt polarity set register
1000B354	EINT_POL_SET5	32	External interrupt polarity set register
1000B380	EINT_POL_CLR0	32	External interrupt polarity clear register
1000B384	EINT_POL_CLR1	32	External interrupt polarity clear register
1000B388	EINT_POL_CLR2	32	External interrupt polarity clear register
1000B38C	EINT_POL_CLR3	32	External interrupt polarity clear register
1000B390	EINT_POL_CLR4	32	External interrupt polarity clear register
1000B394	EINT_POL_CLR5	32	External interrupt polarity clear register
1000B400	EINT_D0EN0	32	Domain 0 external interrupt enable control register
1000B404	EINT_D0EN1	32	Domain 0 external interrupt enable control register
1000B408	EINT_D0EN2	32	Domain 0 external interrupt enable control register
1000B40C	EINT_D0EN3	32	Domain 0 external interrupt enable control register
1000B410	EINT_D0EN4	32	Domain 0 external interrupt enable control register
1000B414	EINT_D0EN5	32	Domain 0 external interrupt enable control register
1000B420	EINT_D1EN0	32	Domain 1 external interrupt enable control register
1000B424	EINT_D1EN1	32	Domain 1 external interrupt enable control register
1000B428	EINT_D1EN2	32	Domain 1 external interrupt enable control register
1000B42C	EINT_D1EN3	32	Domain 1 external interrupt enable control register

Module name: ap_cirq_eint_reg Base address: (+1000B000h)

1000B430	EINT_D1EN4	32	Domain 1 external interrupt enable control register
1000B434	EINT_D1EN5	32	Domain 1 external interrupt enable control register
1000B440	EINT_D2EN0	32	Domain 1 external interrupt enable control register
1000B444	EINT_D2EN1	32	Domain 1 external interrupt enable control register
1000B448	EINT_D2EN2	32	Domain 1 external interrupt enable control register
1000B44C	EINT_D2EN3	32	Domain 1 external interrupt enable control register
1000B450	EINT_D2EN4	32	Domain 1 external interrupt enable control register
1000B454	EINT_D2EN5	32	Domain 1 external interrupt enable control register
1000B500 ~ 1000B50C	EINT_DBNC_3_0[n] (n=0~3)	32	External interrupt debounce control register
1000B600 ~ 1000B60C	EINT_DBNC_SET_3_0[n] (n=0~3)	32	External interrupt debounce control register
1000B700 ~ 1000B70C	EINT_DBNC_CLR_3_0[n] (n=0~3)	32	External interrupt debounce control register

1000B000 **EINT_STA0** **External interrupt status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND0	Each bit read as 1 indicates the corresponding external interrupt is pending

1000B004 **EINT_STA1** **External interrupt status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND1	Each bit read as 1 indicates the corresponding external

Bit(s)	Mnemonic	Name	Description
			interrupt is pending

1000B008 EINT_STA2 External interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND2	Each bit read as 1 indicates the corresponding external interrupt is pending

1000B00C EINT_STA3 External interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND3	Each bit read as 1 indicates the corresponding external interrupt is pending

1000B010 EINT_STA4 External interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND4	Each bit read as 1 indicates the corresponding external interrupt is pending

1000B014 EINT_STA5 External interrupt status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_PEND5[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_PEND5[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_PEND5	Each bit read as 1 indicates the corresponding external interrupt is pending

1000B040 EINT_ACK0 External interrupt acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK0	Write 1 to specific bit acknowledges the corresponding external interrupt

1000B044 EINT_ACK1 External interrupt acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK1	Write 1 to specific bit acknowledges the corresponding external interrupt

1000B048 **EINT_ACK2** **External interrupt acknowledge register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK2	Write 1 to specific bit acknowledges the corresponding external interrupt

1000B04C **EINT_ACK3** **External interrupt acknowledge register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK3	Write 1 to specific bit acknowledges the corresponding external interrupt

1000B050 **EINT_ACK4** **External interrupt acknowledge register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK4	**Write 1 to specific bit acknowledges the corresponding external interrupt**

1000B054 EINT_ACK5 External interrupt acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_ACK4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_ACK4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_ACK4	**Write 1 to specific bit acknowledges the corresponding external interrupt**

1000B080 EINT_MASK0 External interrupt mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_MASK0[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_MASK0[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK0	**External interrupt mask value**

1000B084 EINT_MASK1 External interrupt mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_MASK1[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_MASK1[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK1	External interrupt mask value

1000B088 EINT_MASK2 External interrupt mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_MASK2[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_MASK2[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK2	External interrupt mask value

1000B08C EINT_MASK3 External interrupt mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_MASK3[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_MASK3[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK3	External interrupt mask value

1000B090 EINT_MASK4 External interrupt mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	EINT_MASK4[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	EINT_MASK4[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK4	External interrupt mask value

1000B094 EINT_MASK5 External interrupt mask register 000001FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EINT_MASK5								
Type								RO								
Reset								1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
8:0		EINT_MASK5	External interrupt mask value

1000B0C0 EINT_MASK_SET0 External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET0	Write 1 to specific bit sets the mask of corresponding external interrupt

1000B0C4 EINT_MASK_SET1 External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET1[15:0]															
Type	WO															

1000B0C4 EINT_MASK_SET1 External interrupt mask set register 00000000

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET1	Write 1 to specific bit sets the mask of corresponding external interrupt

1000B0C8 EINT_MASK_SET2 External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET2	Write 1 to specific bit sets the mask of corresponding external interrupt

1000B0CC EINT_MASK_SET3 External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET3	Write 1 to specific bit sets the mask of corresponding external interrupt

1000B0D0 EINT_MASK_SET4 External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

1000B0D0 EINT_MASK_SET4 **External interrupt mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET4	**Write 1 to specific bit sets the mask of corresponding external interrupt**

1000B0D4 EINT_MASK_SET5 **External interrupt mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_SET5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_SET5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_SET5	**Write 1 to specific bit sets the mask of corresponding external interrupt**

1000B100 EINT_MASK_CLR0 **External interrupt mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CL	**Write 1 to specific bit clear the mask of corresponding**

Bit(s)	Mnemonic	Name	Description
		R0	external interrupt

1000B104 **EINT_MASK_CLR1** External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CLR1	Write 1 to specific bit clear the mask of corresponding external interrupt

1000B108 **EINT_MASK_CLR2** External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CLR2	Write 1 to specific bit clear the mask of corresponding external interrupt

1000B10C **EINT_MASK_CLR3** External interrupt mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR3[15:0]															

1000B10C **EINT_MASK_CLR3** **External interrupt mask set register** **00000000**

Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CLR3	Write 1 to specific bit clear the mask of corresponding external interrupt

1000B110 **EINT_MASK_CLR4** **External interrupt mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CLR4	Write 1 to specific bit clear the mask of corresponding external interrupt

1000B114 **EINT_MASK_CLR5** **External interrupt mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_MASK_CLR5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_MASK_CLR5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_MASK_CLR5	Write 1 to specific bit clear the mask of corresponding external interrupt

1000B140 **EINT_SENS0** **External interrupt sensitivity register** **FFFFFFFF**

1000B140 EINT_SENS0 External interrupt sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS0[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS0[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS0	External interrupt sensitivity value

1000B144 EINT_SENS1 External interrupt sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS1[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS1[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS1	External interrupt sensitivity value

1000B148 EINT_SENS2 External interrupt sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS2[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS2[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS2	External interrupt sensitivity value

1000B14C EINT_SENS3 External interrupt sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS3[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS3[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS3	**External interrupt sensitivity value**

1000B150 EINT_SENS4 External interrupt sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS4[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS4[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS4	**External interrupt sensitivity value**

1000B154 EINT_SENS5 External interrupt sensitivity register 000001FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EINT_SENS5								
Type								RO								
Reset								1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
8:0		EINT_SENS5	**External interrupt sensitivity value**

1000B180 **EINT_SENSSET0** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET0	**Write 1 to specific bit sets the sensitivity of corresponding external interrupt**

1000B184 **EINT_SENSSET1** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET1	**Write 1 to specific bit sets the sensitivity of corresponding external interrupt**

1000B188 **EINT_SENSSET2** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET2	Write 1 to specific bit sets the sensitivity of corresponding external interrupt

1000B18C **EINT_SENS_SET3** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET3	Write 1 to specific bit sets the sensitivity of corresponding external interrupt

1000B190 **EINT_SENS_SET4** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET4	Write 1 to specific bit sets the sensitivity of corresponding external interrupt

1000B194 **EINT_SENS_SET5** **External interrupt sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_SET5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	EINT_SENS_SET5[15:0]															

1000B194 EINT_SENS_SET5 External interrupt sensitivity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_SET5	Write 1 to specific bit sets the sensitivity of corresponding external interrupt

1000B1C0 EINT_SENS_CLR0 External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR0	Write 1 to specific bit clear the sensitivity of corresponding external interrupt

1000B1C4 EINT_SENS_CLR1 External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR1	Write 1 to specific bit clear the sensitivity of corresponding external interrupt

1000B1C8 EINT_SENS_CLR2 External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR2	**Write 1 to specific bit clear the sensitivity of corresponding external interrupt**

1000B1CC EINT_SENS_CLR3 External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR3	**Write 1 to specific bit clear the sensitivity of corresponding external interrupt**

1000B1D0 EINT_SENS_CLR4 External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR4	Write 1 to specific bit clear the sensitivity of corresponding external interrupt

1000B1D4 **EINT_SENS_CLR5** External interrupt sensitivity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SENS_CLR5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SENS_CLR5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SENS_CLR5	Write 1 to specific bit clear the sensitivity of corresponding external interrupt

1000B200 **EINT_SOFT0** Software interrupt register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT0	Software interrupt value

1000B204 **EINT_SOFT1** Software interrupt register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT1[15:0]															

1000B204 **EINT_SOFT1** **Software interrupt register** **00000000**

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT1	**Software interrupt value**

1000B208 **EINT_SOFT2** **Software interrupt register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**EINT_SOFT2[31:16]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**EINT_SOFT2[15:0]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT2	**Software interrupt value**

1000B20C **EINT_SOFT3** **Software interrupt register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**EINT_SOFT3[31:16]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**EINT_SOFT3[15:0]**															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT3	**Software interrupt value**

1000B210 **EINT_SOFT4** **Software interrupt register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**EINT_SOFT4[31:16]**															
Type	RO															

1000B210 EINT_SOFT4 Software interrupt register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT4	**Software interrupt value**

1000B214 EINT_SOFT5 Software interrupt register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT5[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT5[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT5	**Software interrupt value**

1000B240 EINT_SOFT_SET0 Software interrupt set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET0	**Write 1 to specific bit sets the corresponding software interrupt**

1000B244 EINT_SOFT_SET1 Software interrupt set register 00000000

1000B244 **EINT_SOFT_SET1** **Software interrupt set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET1	Write 1 to specific bit sets the corresponding software interrupt

1000B248 **EINT_SOFT_SET2** **Software interrupt set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET2	Write 1 to specific bit sets the corresponding software interrupt

1000B24C **EINT_SOFT_SET3** **Software interrupt set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET3	Write 1 to specific bit sets the corresponding software interrupt

1000B250 EINT_SOFT_SET4 Software interrupt set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET4	Write 1 to specific bit sets the corresponding software interrupt

1000B254 EINT_SOFT_SET5 Software interrupt set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_SET5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_SET5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_SET5	Write 1 to specific bit sets the corresponding software interrupt

1000B280 EINT_SOFT_CLR0 Software interrupt clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	EINT_SOFT_CLR0[15:0]															

1000B280 EINT_SOFT_CLR0 **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR0	Write 1 to specific bit clear the corresponding software interrupt

1000B284 EINT_SOFT_CLR1 **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR1	Write 1 to specific bit clear the corresponding software interrupt

1000B288 EINT_SOFT_CLR2 **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR2	Write 1 to specific bit clear the corresponding software interrupt

1000B28C **EINT_SOFT_CLR3** **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR3	Write 1 to specific bit clear the corresponding software interrupt

1000B290 **EINT_SOFT_CLR4** **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR4	Write 1 to specific bit clear the corresponding software interrupt

1000B294 **EINT_SOFT_CLR5** **Software interrupt clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_SOFT_CLR5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_SOFT_CLR5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_SOFT_CLR5	Write 1 to specific bit clear the corresponding software interrupt

1000B300 EINT_POL0 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL0	External interrupt polarity value

1000B304 EINT_POL1 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL1	External interrupt polarity value

1000B308 EINT_POL2 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL2[15:0]															
Type	RO															

1000B308 **EINT_POL2** **External interrupt polarity register** **00000000**

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL2	External interrupt polarity value

1000B30C **EINT_POL3** **External interrupt polarity register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL3	External interrupt polarity value

1000B310 **EINT_POL4** **External interrupt polarity register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL4	External interrupt polarity value

1000B314 **EINT_POL5** **External interrupt polarity register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL5[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

1000B314 EINT_POL5 External interrupt polarity register 00000000

Name	EINT_POL5[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL5	External interrupt polarity value

1000B340 EINT_POL_SET0 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET0	Write 1 to specific bit sets the polarity of corresponding external interrupt

1000B344 EINT_POL_SET1 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET1	Write 1 to specific bit sets the polarity of corresponding external interrupt

1000B348 EINT_POL_SET2 External interrupt polarity set register 00000000

1000B348 **EINT_POL_SET2** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET2	**Write 1 to specific bit sets the polarity of corresponding external interrupt**

1000B34C **EINT_POL_SET3** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET3	**Write 1 to specific bit sets the polarity of corresponding external interrupt**

1000B350 **EINT_POL_SET4** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET4	Write 1 to specific bit sets the polarity of corresponding external interrupt

1000B354 EINT_POL_SET5 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_SET5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_SET5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_SET5	Write 1 to specific bit sets the polarity of corresponding external interrupt

1000B380 EINT_POL_CLR0 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR0	Write 1 to specific bit clear the polarity of corresponding external interrupt

1000B384 EINT_POL_CLR1 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	EINT_POL_CLR1[15:0]															

1000B384 **EINT_POL_CLR1** **External interrupt polarity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR1	Write 1 to specific bit clear the polarity of corresponding external interrupt

1000B388 **EINT_POL_CLR2** **External interrupt polarity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR2	Write 1 to specific bit clear the polarity of corresponding external interrupt

1000B38C **EINT_POL_CLR3** **External interrupt polarity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR3	Write 1 to specific bit clear the polarity of corresponding external interrupt

1000B390 **EINT_POL_CLR4** **External interrupt polarity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR4	**Write 1 to specific bit clear the polarity of corresponding external interrupt**

1000B394 **EINT_POL_CLR5** **External interrupt polarity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_POL_CLR5[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_POL_CLR5[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_POL_CLR5	**Write 1 to specific bit clear the polarity of corresponding external interrupt**

1000B400 **EINT_DOEN0** **Domain 0 external interrupt enable control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN0	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B404 EINT_DOEN1 Domain 0 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN1	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B408 EINT_DOEN2 Domain 0 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN2	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B40C EINT_DOEN3 Domain 0 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	EINT_DOEN3[15:0]															

1000B40C **EINT_DOEN3** **Domain 0 external interrupt enable control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN3	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B410 **EINT_DOEN4** **Domain 0 external interrupt enable control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN4[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN4	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B414 **EINT_DOEN5** **Domain 0 external interrupt enable control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_DOEN5[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_DOEN5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_DOEN5	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B420 EINT_D1EN0 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D1EN0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN0	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B424 EINT_D1EN1 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D1EN1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN1	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B428 EINT_D1EN2 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D1EN2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN2	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B42C EINT_D1EN3 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D1EN3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN3	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B430 EINT_D1EN4 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN4[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D1EN4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN4	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B434 EINT_D1EN5 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D1EN5[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	EINT_D1EN5[15:0]															

1000B434 EINT_D1EN5 Domain 1 external interrupt enable control register 00000000

e																
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D1EN5	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B440 EINT_D2EN0 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN0	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B444 EINT_D2EN1 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN1	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B448 EINT_D2EN2 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN2	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B44C EINT_D2EN3 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN3	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B450 EINT_D2EN4 Domain 1 external interrupt enable control register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN4[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN4	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B454 **EINT_D2EN5** **Domain 1 external interrupt enable control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EINT_D2EN5[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EINT_D2EN5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		EINT_D2EN5	Write 1 to specific bit enables the corresponding software external interrupt in domain 0

1000B500~1000B50C **EINT_DBNC_3_0[n] (n=0~3)** **External interrupt debounce control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBNC_SETTING3					DBNC_RST3	EN3		DBNC_SETTING2					DBNC_RST2	EN2
Type		RO					RO	RO		RO					RO	RO
Reset		0	0	0			0	0		0	0	0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DBNC_SETTING1					DBNC_RST1	EN1		DBNC_SETTING0					DBNC_RST0	EN0
Type		RO					RO	RO		RO					RO	RO
Reset		0	0	0			0	0		0	0	0			0	0

Bit(s)	Mnemonic	Name	Description
30:28		DBNC_SETTING3	Debounce setting 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms

Bit(s)	Mnemonic	Name	Description
			111: 512ms
25		DBNC_RST3	**Debounce counter reset** 0: Negative 1: Positive
24		EN3	**Enable debounce function** 0: Disable 1: Enable
22:20		DBNC_SETTING2	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
17		DBNC_RST2	**Debounce counter reset** 0: Negative 1: Positive
16		EN2	**Enable debounce function** 0: Disable 1: Enable
14:12		DBNC_SETTING1	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
9		DBNC_RST1	**Debounce counter reset** 0: Negative 1: Positive
8		EN1	**Enable debounce function** 0: Disable 1: Enable
6:4		DBNC_SETTING0	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
1		DBNC_RST0	**Debounce counter reset** 0: Negative 1: Positive
0		EN0	**Enable debounce function** 0: Disable 1: Enable

1000B600~1000B60C	EINT_DBNC_SET_3_o[n] (n=0~3)	External interrupt debounce control register	00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBNC_SETTING_SET3					DBNC_RST_SET3	EN_SET3		DBNC_SETTING_SET2					DBNC_RST_SET2	EN_SET2
Type		WO					WO	WO		WO					WO	WO
Reset		0	0	0			0	0		0	0	0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DBNC_SETTING_SET1					DBNC_RST_SET1	EN_SET1		DBNC_SETTING_SETo					DBNC_RST_SETo	EN_SETo
Type		WO					WO	WO		WO					WO	WO
Reset		0	0	0			0	0		0	0	0			0	0

Bit(s)	Mnemonic	Name	Description
30:28		DBNC_SETTING_SET3	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
25		DBNC_RST_SET3	**Debounce counter reset** 0: Negative 1: Positive
24		EN_SET3	**Enable debounce function** 0: Disable 1: Enable
22:20		DBNC_SETTING_SET2	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
17		DBNC_RST_SET2	**Debounce counter reset** 0: Negative 1: Positive
16		EN_SET2	**Enable debounce function** 0: Disable 1: Enable
14:12		DBNC_SETTING	**Debounce setting**

Bit(s)	Mnemonic	Name	Description
		_SET1	000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
9		DBNC_RST_SET1	**Debounce counter reset** 0: Negative 1: Positive
8		EN_SET1	**Enable debounce function** 0: Disable 1: Enable
6:4		DBNC_SETTING_SET0	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
1		DBNC_RST_SET0	**Debounce counter reset** 0: Negative 1: Positive
0		EN_SET0	**Enable debounce function** 0: Disable 1: Enable

1000B700~1000B70C **EINT_DBNC_CLR_3_0[n] (n=0~3)** **External interrupt debounce control register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBNC_SETTING_CLR_CLR3					DBNC_RST3	EN_CLR3		DBNC_SETTING_CLR_CLR2					DBNC_RST2	EN_CLR2
Type		WO					WO	WO		WO					WO	WO
Reset		0	0	0			0	0		0	0	0			0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		DBNC_SETTING_CLR_CLR1					DBNC_RST1	EN_CLR1		DBNC_SETTING_CLR_CLR0					DBNC_RST0	EN_CLR0
Type		WO					WO	WO		WO					WO	WO
Reset		0	0	0			0	0		0	0	0			0	0

Bit(s)	Mnemonic	Name	Description
30:28		DBNC_SETTING	**Debounce setting**

Bit(s)	Mnemonic	Name	Description
		_CLR_CLR3	000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
25		DBNC_RST3	**Debounce counter reset** 0: Negative 1: Positive
24		EN_CLR3	**Enable debounce function** 0: Disable 1: Enable
22:20		DBNC_SETTING_CLR_CLR2	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
17		DBNC_RST2	**Debounce counter reset** 0: Negative 1: Positive
16		EN_CLR2	**Enable debounce function** 0: Disable 1: Enable
14:12		DBNC_SETTING_CLR_CLR1	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
9		DBNC_RST1	**Debounce counter reset** 0: Negative 1: Positive
8		EN_CLR1	**Enable debounce function** 0: Disable 1: Enable
6:4		DBNC_SETTING_CLR_CLR0	**Debounce setting** 000: 0.5ms 001: 1ms 010: 16ms 011: 32ms 100: 64ms 101: 128ms 110: 256ms 111: 512ms
1		DBNC_RST0	**Debounce counter reset** 0: Negative

Bit(s)	Mnemonic	Name	Description
			1: Positive
0		EN_CLR0	**Enable debounce function** 0: Disable 1: Enable

6 System Interrupt Controller

6.1 Introduction

For CA7 processor which has embedded interrupt controllers (GIC), the part of the MCUSYS will need to keep feeding clock and power to make the interrupt functional. However, due to power/leakage overhead introduced by higher clock ratio and deep submicron processes, reserving an always on (or frequently turned on) domain in MCUSYS has become power ineffective. The system interrupt controller (SYS_CIRQ) is a low power interrupt controller is designed to work outside MCUSYS as a second level interrupt controller. With SYS_CIRQ, MCUSYS can be completely turned off to improve the system power consumption without losing interrupts.

6.2 Feature list

SYS_CIRQ supports interrupts which can configure the following attributes individually.

- Polarity inversion
- Edge/level trigger selection

The interrupts will feed through SYS_C and connect to GIC in MCUSYS. When SYS_CIRQ is enabled, it will record the edge-sensitive interrupts and generate a pulse signal to CPU GIC when the flush command is executed.

6.3 Block Diagram

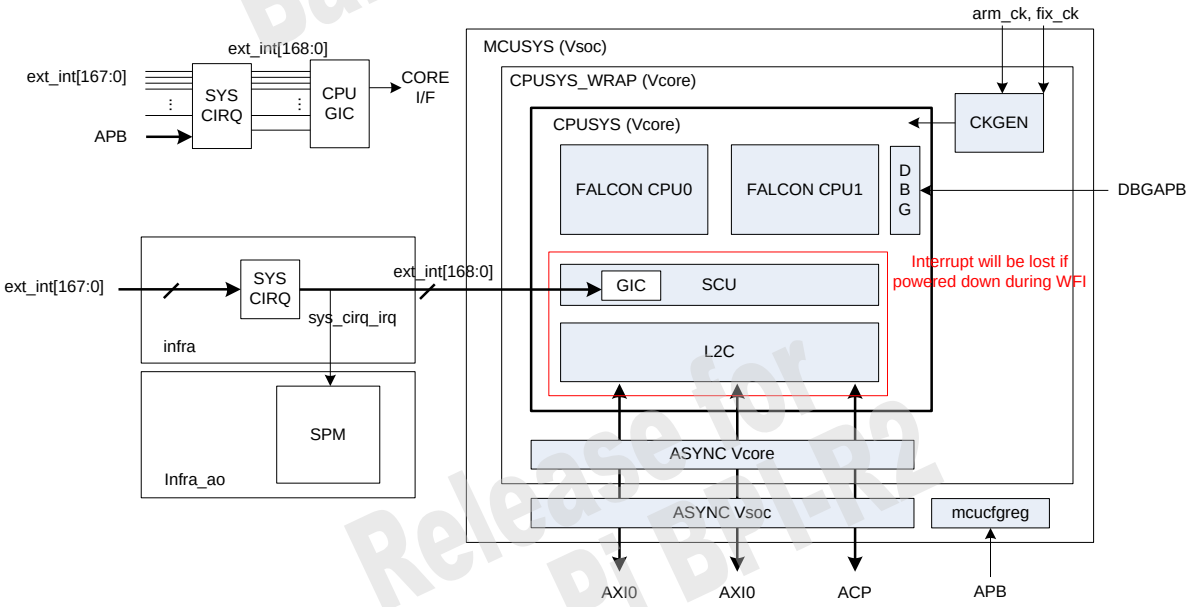


Figure 6-1: System level block diagram of System Interrupt Controller

The SYS_CIRQ controller is integrated in between MCUSYS and other interrupt sources as the second level interrupt controller. All interrupts are fed through SYS_CIRQ controller then bypassed to MCUSYS. In normal mode (where MCUSYS GIC is active), SYS_CIRQ is disabled and interrupts will directly issued to MCUSYS. When MCUSYS enters the sleep mode, where GIC is power downed. SYS_CIRQ controller will be enabled and monitor all edge-trigger interrupts (only edge-triggered interrupt will be lost in this scenario). When an edge-trigger interrupt is triggered, it will be recorded in the SYS_CIRQ_STA register and can be restored to GIC by SW context restore or the SYS_CIRQ flush function.

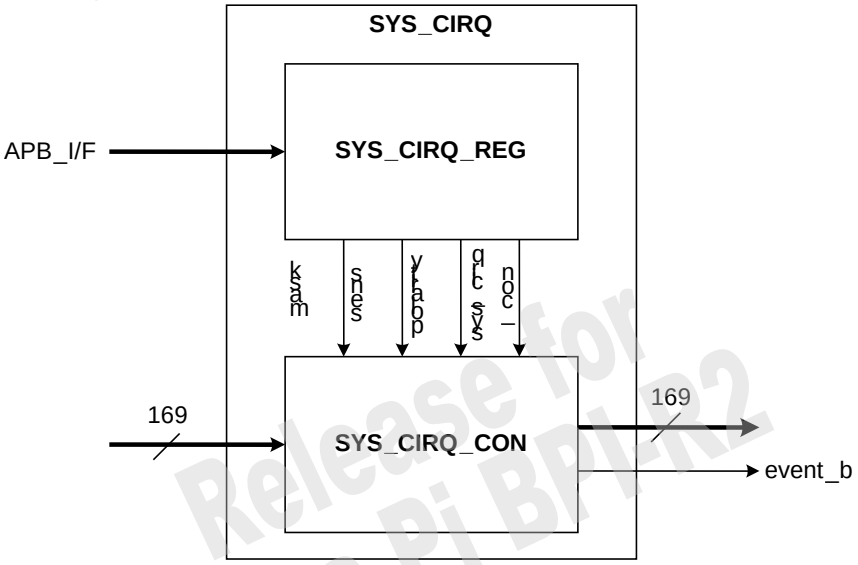


Figure 6-2: Block diagram of system interrupt controller

6.4 Register Definition

Module name: sys_cirq_reg Base address: (+10204000h)

Address	Name	Width	Register Function
10204000	CIRQ_STA0	32	System CIRQ status register
10204004	CIRQ_STA1	32	System CIRQ status register
10204008	CIRQ_STA2	32	System CIRQ status register
1020400C	CIRQ_STA3	32	System CIRQ status register
10204010	CIRQ_STA4	32	System CIRQ status register
10204014	CIRQ_STA5	32	System CIRQ status register
10204040	CIRQ_ACK0	32	System CIRQ acknowledge register
10204044	CIRQ_ACK1	32	System CIRQ acknowledge register
10204048	CIRQ_ACK2	32	System CIRQ acknowledge register
1020404C	CIRQ_ACK3	32	System CIRQ acknowledge register
10204050	CIRQ_ACK4	32	System CIRQ acknowledge register

Module name: sys_cirq_reg Base address: (+10204000h)

10204054	CIRQ_ACK5	32	System CIRQ acknowledge register
10204080	CIRQ_MASK0	32	System CIRQ mask register
10204084	CIRQ_MASK1	32	System CIRQ mask register
10204088	CIRQ_MASK2	32	System CIRQ mask register
1020408C	CIRQ_MASK3	32	System CIRQ mask register
10204090	CIRQ_MASK4	32	System CIRQ mask register
10204094	CIRQ_MASK5	32	System CIRQ mask register
102040C0	CIRQ_MASK_SET0	32	System CIRQ mask set register
102040C4	CIRQ_MASK_SET1	32	System CIRQ mask set register
102040C8	CIRQ_MASK_SET2	32	System CIRQ mask set register
102040CC	CIRQ_MASK_SET3	32	System CIRQ mask set register
102040D0	CIRQ_MASK_SET4	32	System CIRQ mask set register
102040D4	CIRQ_MASK_SET5	32	System CIRQ mask set register
10204100	CIRQ_MASK_CLR0	32	System CIRQ mask set register
10204104	CIRQ_MASK_CLR1	32	System CIRQ mask set register
10204108	CIRQ_MASK_CLR2	32	System CIRQ mask set register
1020410C	CIRQ_MASK_CLR3	32	System CIRQ mask set register
10204110	CIRQ_MASK_CLR4	32	System CIRQ mask set register
10204114	CIRQ_MASK_CLR5	32	System CIRQ mask set register
10204140	CIRQ_SENS0	32	System CIRQ sensitivity register
10204144	CIRQ_SENS1	32	System CIRQ sensitivity register
10204148	CIRQ_SENS2	32	System CIRQ sensitivity register
1020414C	CIRQ_SENS3	32	System CIRQ sensitivity register
10204150	CIRQ_SENS4	32	System CIRQ sensitivity register
10204154	CIRQ_SENS5	32	System CIRQ sensitivity register
10204180	CIRQ_SENS_SET0	32	System CIRQ sensitivity set register
10204184	CIRQ_SENS_SET1	32	System CIRQ sensitivity set register
10204188	CIRQ_SENS_SET2	32	System CIRQ sensitivity set register
1020418C	CIRQ_SENS_SET3	32	System CIRQ sensitivity set register
10204190	CIRQ_SENS_SET4	32	System CIRQ sensitivity set register
10204194	CIRQ_SENS_SET5	32	System CIRQ sensitivity set register
102041C0	CIRQ_SENS_CLR0	32	System CIRQ sensitivity clear register

Module name: sys_cirq_reg Base address: (+10204000h)

102041C4	CIRQ_SENS_CLR1	32	System CIRQ sensitivity clear register
102041C8	CIRQ_SENS_CLR2	32	System CIRQ sensitivity clear register
102041CC	CIRQ_SENS_CLR3	32	System CIRQ sensitivity clear register
102041D0	CIRQ_SENS_CLR4	32	System CIRQ sensitivity clear register
102041D4	CIRQ_SENS_CLR5	32	System CIRQ sensitivity clear register
10204200	CIRQ_POL0	32	External interrupt polarity register
10204204	CIRQ_POL1	32	External interrupt polarity register
10204208	CIRQ_POL2	32	External interrupt polarity register
1020420C	CIRQ_POL3	32	External interrupt polarity register
10204210	CIRQ_POL4	32	External interrupt polarity register
10204214	CIRQ_POL5	32	External interrupt polarity register
10204240	CIRQ_POL_SET0	32	External interrupt polarity set register
10204244	CIRQ_POL_SET1	32	External interrupt polarity set register
10204248	CIRQ_POL_SET2	32	External interrupt polarity set register
1020424C	CIRQ_POL_SET3	32	External interrupt polarity set register
10204250	CIRQ_POL_SET4	32	External interrupt polarity set register
10204254	CIRQ_POL_SET5	32	External interrupt polarity set register
10204280	CIRQ_POL_CLR0	32	External interrupt polarity clear register
10204284	CIRQ_POL_CLR1	32	External interrupt polarity clear register
10204288	CIRQ_POL_CLR2	32	External interrupt polarity clear register
1020428C	CIRQ_POL_CLR3	32	External interrupt polarity clear register
10204290	CIRQ_POL_CLR4	32	External interrupt polarity clear register
10204294	CIRQ_POL_CLR5	32	External interrupt polarity clear register
10204300	CIRQ_CON	32	Sytem CIRQ control register

10204000 CIRQ_STA0 System CIRQ status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CIRQ_PEND0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	CIRQ_PEND0[15:0]															

10204000 CIRQ_STA0 System CIRQ status register 00000000

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_PEND0	Each bit read as 1 indicates the corresponding system CIRQ is pending

10204004 CIRQ_STA1 System CIRQ status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CIRQ_PEND1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	CIRQ_PEND1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_PEND1	Each bit read as 1 indicates the corresponding system CIRQ is pending

10204008 CIRQ_STA2 System CIRQ status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CIRQ_PEND2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	CIRQ_PEND2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_PEND2	Each bit read as 1 indicates the corresponding system CIRQ is pending

1020400C CIRQ_STA3 System CIRQ status register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CIRQ_PEND3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	CIRQ_PEND3[15:0]															
Type	RO															

10204004C **CIRQ_STA3** **System CIRQ status register** **00000000**

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_PEND3	Each bit read as 1 indicates the corresponding system CIRQ is pending

10204010 **CIRQ_STA4** **System CIRQ status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CIRQ_PEND4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	CIRQ_PEND4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_PEND4	Each bit read as 1 indicates the corresponding system CIRQ is pending

10204014 **CIRQ_STA5** **System CIRQ status register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_STA5								
Type								RO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_STA5	Each bit read as 1 indicates the corresponding system CIRQ is pending

10204040 **CIRQ_ACK0** **System CIRQ acknowledge register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_ACK0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

10204040 CIRQ_ACK0 System CIRQ acknowledge register 00000000

Name	CIRQ_ACK0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_ACK0	Write 1 to specific bit acknowledges the corresponding system CIRQ

10204044 CIRQ_ACK1 System CIRQ acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_ACK1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_ACK1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_ACK1	Write 1 to specific bit acknowledges the corresponding system CIRQ

10204048 CIRQ_ACK2 System CIRQ acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_ACK2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_ACK2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_ACK2	Write 1 to specific bit acknowledges the corresponding system CIRQ

1020404C CIRQ_ACK3 System CIRQ acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_ACK3[31:16]															

1020404C CIRQ_ACK3 System CIRQ acknowledge register 00000000

Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_ACK3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_ACK3	Write 1 to specific bit acknowledges the corresponding system CIRQ

10204050 CIRQ_ACK4 System CIRQ acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_ACK4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_ACK4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_ACK4	Write 1 to specific bit acknowledges the corresponding system CIRQ

10204054 CIRQ_ACK5 System CIRQ acknowledge register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_ACK5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_ACK5	Write 1 to specific bit acknowledges the corresponding system CIRQ

10204080 CIRQ_MASK0 System CIRQ mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK0[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK0[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK0	**system CIRQ mask value**

10204084 CIRQ_MASK1 System CIRQ mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK1[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK1[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK1	**system CIRQ mask value**

10204088 CIRQ_MASK2 System CIRQ mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK2[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK2[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK2	**system CIRQ mask value**

1020408C CIRQ_MASK3 System CIRQ mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK3[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK3[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK3	system CIRQ mask value

10204090 CIRQ_MASK4 System CIRQ mask register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK4[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK4[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK4	system CIRQ mask value

10204094 CIRQ_MASK5 System CIRQ mask register 000000FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_MASK5								
Type								RO								
Reset								0	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_MASK5	system CIRQ mask value

102040C0 CIRQ_MASK_SET0 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_SET0	**Write 1 to specific bit sets the mask of corresponding system CIRQ**

102040C4 CIRQ_MASK_SET1 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_SET1	**Write 1 to specific bit sets the mask of corresponding system CIRQ**

102040C8 CIRQ_MASK_SET2 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_SET2	Write 1 to specific bit sets the mask of corresponding system CIRQ

102040CC **CIRQ_MASK_SET3** **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_SET3	Write 1 to specific bit sets the mask of corresponding system CIRQ

102040D0 **CIRQ_MASK_SET4** **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_SET4	Write 1 to specific bit sets the mask of corresponding system CIRQ

102040D4 **CIRQ_MASK_SET5** **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam								CIRQ_MASK_SET5								

102040D4 CIRQ_MASK_SET5 **System CIRQ mask set register** **00000000**

e																
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_MASK_SET5	**Write 1 to specific bit sets the mask of corresponding system CIRQ**

10204100 CIRQ_MASK_CLR0 **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**CIRQ_MASK_CLR0[31:16]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CIRQ_MASK_CLR0[15:0]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_CLR0	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

10204104 CIRQ_MASK_CLR1 **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**CIRQ_MASK_CLR1[31:16]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**CIRQ_MASK_CLR1[15:0]**															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_CLR1	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

10204108 CIRQ_MASK_CLR2 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_CLR2	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

1020410C CIRQ_MASK_CLR3 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_CLR3	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

10204110 CIRQ_MASK_CLR4 System CIRQ mask set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_MASK_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_MASK_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_MASK_CLR4	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

10204114 **CIRQ_MASK_CLR5** **System CIRQ mask set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_MASK_CLR5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_MASK_CLR5	**Write 1 to specific bit clear the mask of corresponding system CIRQ**

10204140 **CIRQ_SENS0** **System CIRQ sensitivity register** **FFFFFFE3**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS0[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS0[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS0	**system CIRQ sensitivity value**

10204144 **CIRQ_SENS1** **System CIRQ sensitivity register** **FEFC7FFF**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS1[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS1[15:0]															
Type	RO															

10204144 CIRQ_SENS1 System CIRQ sensitivity register FEFC7FFF

Reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS1	system CIRQ sensitivity value

10204148 CIRQ_SENS2 System CIRQ sensitivity register FFEFDFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS2[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS2[15:0]															
Type	RO															
Reset	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS2	system CIRQ sensitivity value

1020414C CIRQ_SENS3 System CIRQ sensitivity register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS3[31:16]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS3[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS3	system CIRQ sensitivity value

10204150 CIRQ_SENS4 System CIRQ sensitivity register 0777FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS4[31:16]															
Type	RO															
Reset	0	0	0	0	0	1	1	1	0	1	1	1	0	1	1	1

10204150 **CIRQ_SENS4** **System CIRQ sensitivity register** **0777FFFF**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS4[15:0]															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS4	system CIRQ sensitivity value

10204154 **CIRQ_SENS5** **System CIRQ sensitivity register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_SENS5								
Type								RO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_SENS5	system CIRQ sensitivity value

10204180 **CIRQ_SENS_SET0** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_SET0	Write 1 to specific bit sets the sensitivity of corresponding system CIRQ

10204184 **CIRQ_SENS_SET1** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_SET1	**Write 1 to specific bit sets the sensitivity of corresponding system CIRQ**

10204188 **CIRQ_SENS_SET2** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_SET2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_SET2	**Write 1 to specific bit sets the sensitivity of corresponding system CIRQ**

1020418C **CIRQ_SENS_SET3** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_SET3	Write 1 to specific bit sets the sensitivity of corresponding system CIRQ

10204190 **CIRQ_SENS_SET4** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_SET4	Write 1 to specific bit sets the sensitivity of corresponding system CIRQ

10204194 **CIRQ_SENS_SET5** **System CIRQ sensitivity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_SENS_SET5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_SENS_SET5	Write 1 to specific bit sets the sensitivity of corresponding system CIRQ

102041C0 **CIRQ_SENS_CLR0** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	CIRQ_SENS_CLR0[15:0]															

102041C0 **CIRQ_SENS_CLR0** **System CIRQ sensitivity clear register** **00000000**

e																
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_CLR0	Write 1 to specific bit clear the sensitivity of corresponding system CIRQ

102041C4 **CIRQ_SENS_CLR1** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_CLR1	Write 1 to specific bit clear the sensitivity of corresponding system CIRQ

102041C8 **CIRQ_SENS_CLR2** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_CLR2	Write 1 to specific bit clear the sensitivity of corresponding system CIRQ

102041CC **CIRQ_SENS_CLR3** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_CLR3	**Write 1 to specific bit clear the sensitivity of corresponding system CIRQ**

102041D0 **CIRQ_SENS_CLR4** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_SENS_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_SENS_CLR4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_SENS_CLR4	**Write 1 to specific bit clear the sensitivity of corresponding system CIRQ**

102041D4 **CIRQ_SENS_CLR5** **System CIRQ sensitivity clear register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_SENS_CLR5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_SENS_CLR5	**Write 1 to specific bit clear the sensitivity of corresponding system CIRQ**

10204200 CIRQ_POL0 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL0	**system CIRQ polarity value**

10204204 CIRQ_POL1 External interrupt polarity register 00038000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL1[15:0]															
Type	RO															
Reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL1	**system CIRQ polarity value**

10204208 CIRQ_POL2 External interrupt polarity register 00102001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL2[15:0]															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL2	system CIRQ polarity value

1020420C CIRQ_POL3 External interrupt polarity register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL3	system CIRQ polarity value

10204210 CIRQ_POL4 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL4	system CIRQ polarity value

10204214 CIRQ_POL5 External interrupt polarity register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_POL5								
Type								RO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_POL5	system CIRQ polarity value

10204240 **CIRQ_POL_SET0** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_SET0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_SET0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_SET0	Write 1 to specific bit sets the polarity of corresponding system CIRQ

10204244 **CIRQ_POL_SET1** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_SET1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_SET1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_SET1	Write 1 to specific bit sets the polarity of corresponding system CIRQ

10204248 **CIRQ_POL_SET2** **External interrupt polarity set register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_SET2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_SET2[15:0]															

10204248 CIRQ_POL_SET2 External interrupt polarity set register 00000000

Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_SET2	Write 1 to specific bit sets the polarity of corresponding system CIRQ

1020424C CIRQ_POL_SET3 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_SET3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_SET3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_SET3	Write 1 to specific bit sets the polarity of corresponding system CIRQ

10204250 CIRQ_POL_SET4 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_SET4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_SET4[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_SET4	Write 1 to specific bit sets the polarity of corresponding system CIRQ

10204254 CIRQ_POL_SET5 External interrupt polarity set register 00000000

10204254 CIRQ_POL_SET5 External interrupt polarity set register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_POL_SET5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_POL_SET5	**Write 1 to specific bit sets the polarity of corresponding system CIRQ**

10204280 CIRQ_POL_CLR0 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_CLR0[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_CLR0[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_CLR0	**Write 1 to specific bit clear the polarity of corresponding system CIRQ**

10204284 CIRQ_POL_CLR1 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_CLR1[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_CLR1[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_CLR1	Write 1 to specific bit clear the polarity of corresponding system CIRQ

10204288 CIRQ_POL_CLR2 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_CLR2[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_CLR2[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_CLR2	Write 1 to specific bit clear the polarity of corresponding system CIRQ

1020428C CIRQ_POL_CLR3 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_CLR3[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CIRQ_POL_CLR3[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_CLR3	Write 1 to specific bit clear the polarity of corresponding system CIRQ

10204290 CIRQ_POL_CLR4 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_POL_CLR4[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nam	CIRQ_POL_CLR4[15:0]															

10204290 CIRQ_POL_CLR4 External interrupt polarity clear register 00000000

e																
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CIRQ_POL_CLR4	**Write 1 to specific bit clear the polarity of corresponding system CIRQ**

10204294 CIRQ_POL_CLR5 External interrupt polarity clear register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								CIRQ_POL_CLR5								
Type								WO								
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0		CIRQ_POL_CLR5	**Write 1 to specific bit clear the polarity of corresponding system CIRQ**

10204300 CIRQ_CON Sytem CIRQ control register 80000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CIRQ_EVENT_B															
Type	RO															
Reset	1															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														CIRQ_FLUSH	CIRQ_EDGE_ONLY	CIRQ_EN
Type														WO	RW	RW
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31		CIRQ_EVENT_B	Indicate sys_cirq_irq_b is triggered
2		CIRQ_FLUSH	Flush pending interrupts
1		CIRQ_EDGE_ONLY	Set edge-only mode, only edge-triggered interrupt will be recorded
0		CIRQ_EN	Enable bit of system CIRQ controller

6.5 Programming Guide

6.5.1 MCUSYS MTCMOS Sequence

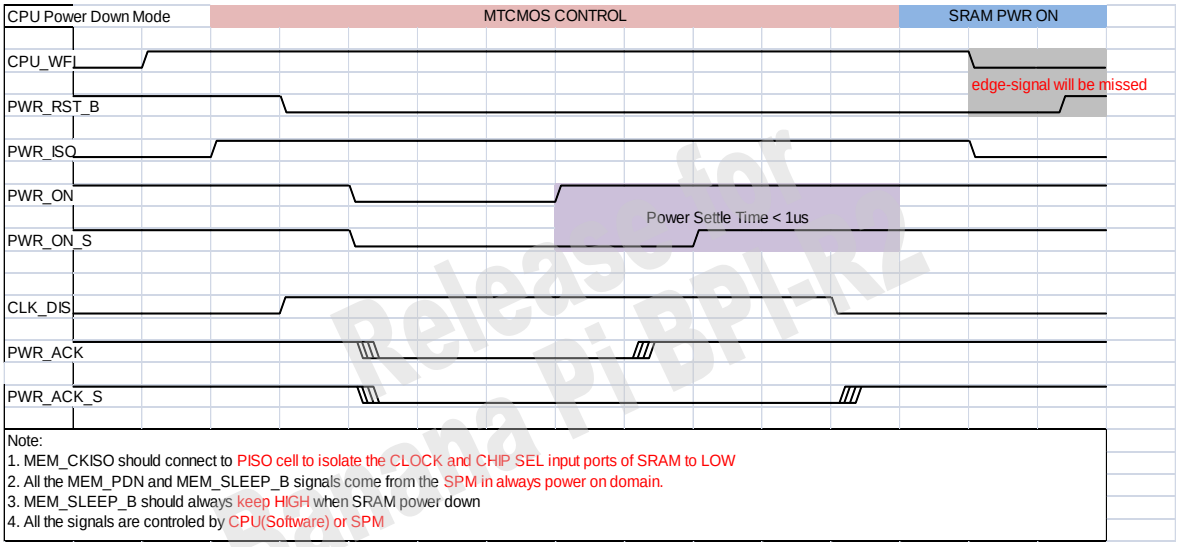


Figure 6-3: MCUSYS MTCMOS Sequence

6.5.2 SW Flow

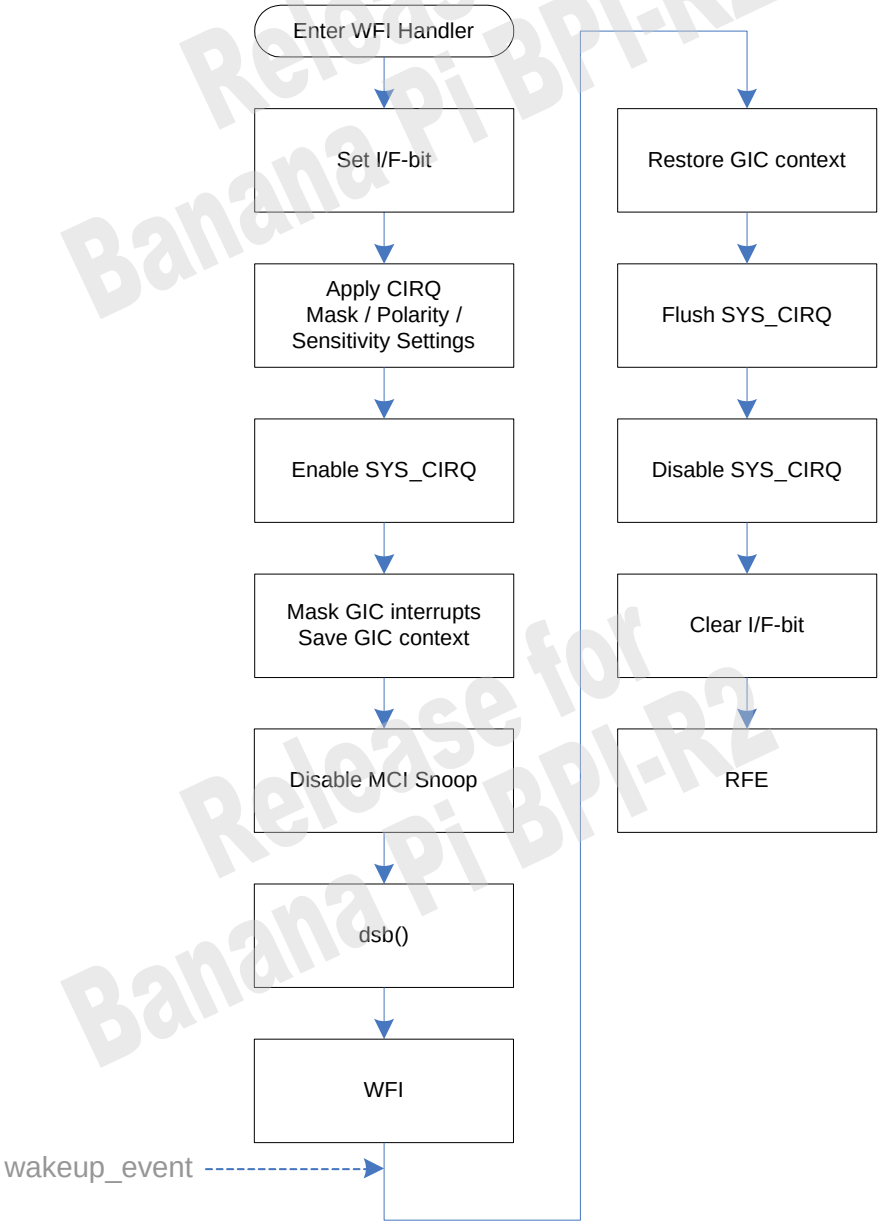


Figure 6-4: Software flow of MCUSYS MTCMOS Sequnce

7 General-Purpose Timer

7.1 Introduction

The General-Purpose Timer (GPT) includes 8 32-bit timers, one 64-bit timer and one secure 32-bit timer. Each timer has 4 operation modes, which are ONE-SHOT, REPEAT, KEEP-GO and FREERUN, and can operate on one of the 2 clock sources, RTC clock (32.768kHz) and system clock (13MHz).

7.2 Feature list

The 4 operation modes for GPT are ONE-SHOT, REPEAT, KEEP-GO and FREERUN. See table below for the functions of each mode.

Table 7-1: Operation mode of GPT

Mode	Auto Stop	Interrupt	Increases when EN=1 and …	When COUNTn equals COMPAREn	Example: Compare is set to 2 **Bold means interrupt*
ONE-SHOT	Yes	Yes	Stops when COUNTn equals to COMPAREn	EN is reset to 0	0,1,2,2,2,2,2,2,2,2,2,…
REPEAT	No	Yes		Count is reset to 0	0,1,2,0,1,2,0,1,2,0,1,2…
KEEP-GO	No	Yes	Reset to 0 when overflow		0,1,2,3,4,5,6,7,8,9,10,…
FREERUN	No	No	Reset to 0 when overflow		0,1,2,3,4,5,6,7,8,9,10,…

Each timer can be programmed to select the clock source, RTC clock (32.76kHz) or system clock (13MHz). After the clock source is determined, the division ratio of the selected clock can be programmed. The division ratio can be fine-granulated as 1, 2, 3, 4 to 13 and coarse-granulated as 16, 32 and 64.

7.3 Block Diagram

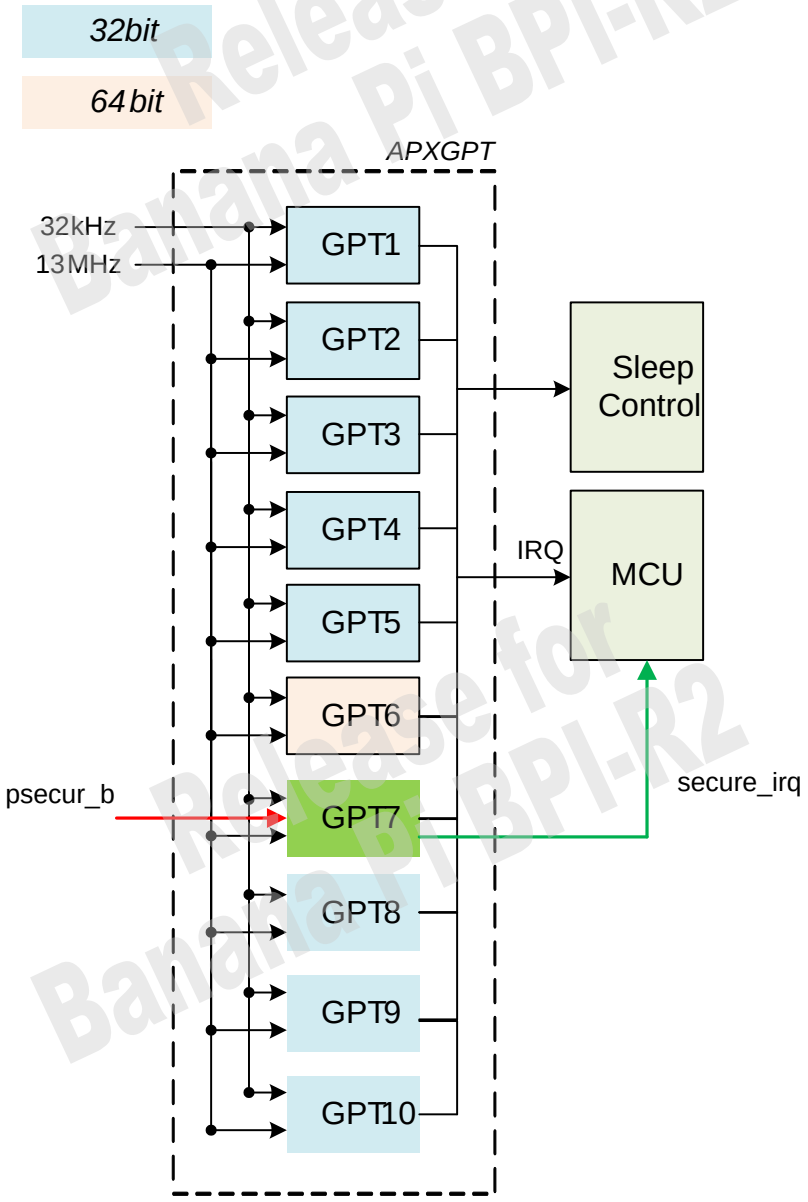


Figure 7-1: Block diagram of GPT

7.4 Register Definition

Module name: APXGPT base address: (+10008000h)

Address	Name	Width	Register function
10008000	**GPT_IRQEN**	32	**GPT IRQ enabling** Controls the enabling/disabling of GPT interrupt
10008004	**GPT_IRQSTA**	32	**GPT IRQ status**

Address	Name	Width	Register function
			Shows the interrupt status of each GPT
10008008	**GPT_IRQACK**	32	**GPT IRQ acknowledgement** Acknowledges the GPT interrupt
10008010	**GPT1_CON**	32	**GPT1 control** The general control for GPT1
10008014	**GPT1_CLK**	32	**GPT1 clock setting** Controls the clock source and division ratio of GPT clock
10008018	**GPT1_COUNT**	32	**GPT1 counter** The timer count of GPT1
1000801C	**GPT1_COMPARE**	32	**GPT1 compare value** The compare value for GPT1
10008020	**GPT2_CON**	32	**GPT2 control** The general control for GPT2
10008024	**GPT2_CLK**	32	**GPT2 clock setting** Controls the clock source and division ratio of GPT clock
10008028	**GPT2_COUNT**	32	**GPT2 counter** The timer count of GPT2
1000802C	**GPT2_COMPARE**	32	**GPT2 compare value** The compare value for GPT2
10008030	**GPT3_CON**	32	**GPT3 control** The general control for GPT3
10008034	**GPT3_CLK**	32	**GPT3 clock setting** Controls the clock source and division ratio of GPT clock
10008038	**GPT3_COUNT**	32	**GPT3 counter** The timer count of GPT3
1000803C	**GPT3_COMPARE**	32	**GPT3 compare value** The compare value for GPT3
10008040	**GPT4_CON**	32	**GPT4 control** The general control for GPT4
10008044	**GPT4_CLK**	32	**GPT4 clock setting** Controls the clock source and division ratio of GPT clock
10008048	**GPT4_COUNT**	32	**GPT4 counter** The timer count of GPT4
1000804C	**GPT4_COMPARE**	32	**GPT4 compare value** The compare value for GPT4
10008050	**GPT5_CON**	32	**GPT5 control** The general control for GPT5
10008054	**GPT5_CLK**	32	**GPT5 clock setting** Controls the clock source and division ratio of GPT

Address	Name	Width	Register function
			clock
10008058	**GPT5_COUNT**	32	**GPT5 counter** The timer count of GPT5
1000805C	**GPT5_COMPARE**	32	**GPT5 compare value** The compare value for GPT5
10008060	**GPT6_CON**	32	**GPT6 control** The general control for GPT6
10008064	**GPT6_CLK**	32	**GPT6 clock setting** Controls the clock source and division ratio of GPT clock
10008068	**GPT6_COUNTL**	32	**GPT6 counter L** The lower word timer count for GPT6
1000806C	**GPT6_COMPAREL**	32	**GPT6 compare value L** The lower word compare value for GPT6
10008078	**GPT6_COUNTH**	32	**GPT6 counter H** The higher word timer count for GPT6
1000807C	**GPT6_COMPAREH**	32	**GPT6 compare value H** The higher word compare value for GPT6
1000808C	**GPT7_CON**	32	**GPT7(secure) control** The general control for GPT7
10008090	**GPT7_CLK**	32	**GPT7(secure) clock setting** Controls the clock source and division ratio of GPT7 clock
10008094	**GPT7_COUNT**	32	**GPT7(secure) counter** The timer count of GPT7
10008098	**GPT7_COMPARE**	32	**GPT7(secure) compare value** The compare value for GPT7
1000809C	**GPT7_SECURE**	32	**GPT7(secure) secure control** The secure control for GPT7
100080a0	**GPT7_IRQEN_SECURE**	32	**GPT7(secure) IRQ enabling** Controls the enabling/disabling of GPT7 interrupt
100080a4	**GPT7_IRQACK_SECURE**	32	**GPT7(secure) IRQ acknowledgement** Acknowledges the GPT7 interrupt
100080a8	**GPT8_CON**	32	**GPT8 control** The general control for GPT8
100080ac	**GPT8_CLK**	32	**GPT8 clock setting** Controls the clock source and division ratio of GPT clock
100080b0	**GPT8_COUNTH**	32	**GPT8 counter** The timer count of GPT8
100080b4	**GPT8_COMPAREH**	32	**GPT8 compare value** The compare value for GPT8
100080b8	**GPT9_CON**	32	**GPT9 control**

Address	Name	Width	Register function
			The general control for GPT9
100080bc	**GPT9_CLK**	32	**GPT9 clock setting** Controls the clock source and division ratio of GPT clock
100080c0	**GPT9_COUNT**	32	**GPT9 counter** The timer count of GPT9
100080c4	**GPT9_COMPARE**	32	**GPT9 compare value** The compare value for GPT9
100080c8	**GPT10_CON**	32	**GPT10 control** The general control for GPT10
100080cc	**GPT10_CLK**	32	**GPT10 clock setting** Controls the clock source and division ratio of GPT clock
100080d0	**GPT10_COUNT**	32	**GPT10 counter** The timer count of GPT10
100080d4	**GPT10_COMPARE**	32	**GPT10 compare value** The compare value for GPT10

10008000 GPT_IRQEN GPT IRQ Enabling 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						**IRQEN**					**IRQEN**					
Type						RW					RW					
Reset						0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
5:0 10:8	**IRQEN**	IRQEN	**Enables interrupt of each GPT** 0: Disable associated interrupt of GPT 1: Enable associated interrupt of GPT

10008004 GPT_IRQSTA GPT IRQ Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						**IRQSTA**					**IRQSTA**					
Type						RU					RU					
Reset						0	0	0	0		0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
5:0	**IRQSTA**	IRQSTA	**Interrupt status of each GPT**

Bit(s)	Mnemonic	Name	Description
10:7			0: No associated interrupt is generated. 1: Associated interrupt is pending and waiting for service.

10008008 GPT_IRQACK GPT IRQ Acknowledgement 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						IRQACK					IRQACK					
Type						WO					WO					
Reset						0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
5:0 10:8	**IRQACK**	IRQACK	**Interrupt acknowledgement for each GPT** 0: No effect 1: Associated interrupt request is acknowledged and should be relinquished.

10008010 GPT1_CON GPT1 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE1				CLR1	EN1
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE1**	MODE1	**Operation mode of GPT1** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR1**	CLR1	**Clears the counter of GPT1 to 0** 0: No effect 1: Clear
0	**EN1**	EN1	**Enables GPT1** 0: Disable 1: Enable

10008014 GPT1_CLK GPT1 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																

10008014 GPT1_CLK GPT1 Clock Setting 00000000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK1	CLKDIV1			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK1**	CLK1	**Sets up clock source of GPT1** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV1**	CLKDIV1	**Setting of GPT1 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008018 GPT1_COUNT GPT1 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER1[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER1[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER1**	COUNTER1	Timer counter of GPT1

1000801C GPT1_COMPARE GPT1 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

1000801C GPT1_COMPARE GPT1 Compare Value 00000000

Name	COMPARE1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE1**	COMPARE1	Compare value of GPT1

10008020 GPT2_CON GPT2 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											**MODE2**				**CLR2**	**EN2**
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE2**	MODE2	**Operation mode of GPT2** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR2**	CLR2	**Clears the counter of GPT2 to 0** 0: No effect 1: Clear
0	**EN2**	EN2	**Enables GPT2** 0: Disable 1: Enable

10008024 GPT2_CLK GPT2 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												**CLK2**	**CLKDIV2**			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK2**	CLK2	**Sets up clock source of GPT2** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV2**	CLKDIV2	**Setting of GPT2 input clock frequency divider** 0000: Clock source divided by 1

Bit(s)	Mnemonic	Name	Description
			0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008028 GPT2_COUNT GPT2 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER2[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER2[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER2**	COUNTER2	Timer counter of GPT2

1000802C GPT2_COMPARE GPT2 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE2**	COMPARE2	Compare value of GPT2

10008030 GPT3_CON GPT3 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																

10008030 GPT3_CON GPT3 Control 00000000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE3				CLR3	EN3
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE3**	MODE3	**Operation mode of GPT3** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR3**	CLR3	**Clears the counter of GPT3 to 0** 0: No effect 1: Clear
0	**EN3**	EN3	**Enables GPT3** 0: Disable 1: Enable

10008034 GPT3_CLK GPT3 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK3	CLKDIV3			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK3**	CLK3	**Sets up clock source of GPT3** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV3**	CLKDIV3	**Setting of GPT3 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16

Bit(s)	Mnemonic	Name	Description
			1110: Clock source divided by 32 1111: Clock source divided by 64

10008038 GPT3_COUNT GPT3 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER3[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER3[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER3**	COUNTER3	Timer counter of GPT3

1000803C GPT3_COMPARE GPT3 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE3[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE3**	COMPARE3	Compare value of GPT3

10008040 GPT4_CON GPT4 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE4				CLR4	EN4
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE4**	MODE4	**Operation mode of GPT4** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR4**	CLR4	**Clears the counter of GPT4 to 0**

Bit(s)	Mnemonic	Name	Description
			0: No effect 1: Clear
0	**EN4**	EN4	**Enables GPT4** 0: Disable 1: Enable

10008044 GPT4_CLK GPT4 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK4	CLKDIV4			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK4**	CLK4	**Sets up clock source of GPT4** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV4**	CLKDIV4	**Setting of GPT4 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008048 GPT4_COUNT GPT4 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER4[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER4[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER4**	COUNTER4	Timer counter of GPT4

1000804C GPT4_COMPARE GPT4 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE4[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE4**	COMPARE4	Compare value of GPT4

10008050 GPT5_CON GPT5 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE5				CLR5	EN5
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE5**	MODE5	**Operation mode of GPT5** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR5**	CLR5	**Clears the counter of GPT5 to 0** 0: No effect 1: Clear
0	**EN5**	EN5	**Enables GPT5** 0: Disable 1: Enable

10008054 GPT5_CLK GPT5 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK5	CLKDIV5			
Type												RW	RW			

10008054 GPT5_CLK GPT5 Clock Setting 00000000

Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK5**	CLK5	**Sets up clock source of GPT5** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV5**	CLKDIV5	**Setting of GPT5 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008058 GPT5_COUNT GPT5 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER5[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER5[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER5**	COUNTER5	Timer counter of GPT5

1000805C GPT5_COMPARE GPT5 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE5[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE5**	COMPARE5	Compare value of GPT5

10008060 **GPT6_CON** **GPT6 Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											**MODE6**				**CLR6**	**EN6**
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE6**	MODE6	**Operation mode of GPT6** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR6**	CLR6	**Clears the counter of GPT6 to 0** 0: No effect 1: Clear
0	**EN6**	EN6	**Enable the GPT6** 0: Disable 1: Enable

10008064 **GPT6_CLK** **GPT6 Clock Setting** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												**CLK6**	**CLKDIV6**			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK6**	CLK6	**Sets up clock source of GPT6** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV6**	CLKDIV6	**GPT6 input clock frequency divider setting** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7

Bit(s)	Mnemonic	Name	Description
			0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008068 GPT6_COUNTL GPT6 Counter L 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER6L[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER6L[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER6L**	COUNTER6L	**Lower word of timer count of GPT6** A read operation of GPT6_COUNTL will make GPT6_COUNTH fixed until the next read operation of GPT6_COUNTL.

1000806C GPT6_COMPAREL GPT6 Compare Value L 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE6L[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE6L[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE6L**	COMPARE6L	Higher word of timer count of GPT6

10008078 GPT6_COUNTH GPT6 Counter H 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER6H[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER6H[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER6H**	COUNTER6H	Lower word of compare value of GPT6

1000807C GPT6_COMPAREH GPT6 Compare Value H 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE6H[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE6H[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE6H**	COMPARE6H	Higher word of compare of GPT6

1000808C GPT7_CON GPT7 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE7				CLR7	EN7
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE7**	MODE7	**Operation mode of GPT7** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR7**	CLR7	**Clears the counter of GPT7 to 0** 0: No effect 1: Clear
0	**EN7**	EN7	**Enables GPT7** 0: Disable 1: Enable

10008090 GPT7_CLK GPT7 Clock Setting 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK7	CLKDIV7			

10008090 GPT7_CLK **GPT7 Clock Setting** **00000000**

Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK7**	CLK7	**Sets up clock source of GPT7** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV7**	CLKDIV7	**Setting of GPT7 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

10008094 GPT7_COUNT **GPT7 Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**COUNTER7[31:16]**															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**COUNTER7[15:0]**															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER7**	COUNTER7	Timer counter of GPT7

10008098 GPT7_COMPARE **GPT7 Compare Value** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**COMPARE7[31:16]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**COMPARE7[15:0]**															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE7**	COMPARE7	Compare value of GPT7

1000809C GPT7_SECURE **GPT7 Secure control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**Secure_on**
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**GPT7_SECURE**	SECURE_ON	**Secure control of GPT7** 0: normol mode 1: secure mode

100080a0 GPT7_IRQEN_SECURE **GPT7 IRQ_enabling** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**Irqen_secur**
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**GPT7_IRQEN_SECUR**	IRQEN_SECURE	**Enable irq GPT7(secure)** 0: Disable associated interrupt of GPT7 1: Enable associated interrupt of GPT7

100080a4 GPT7_IRQACK_SECURE **GPT7 IRQ Acknowledgement** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**ack_secure**
Type																WO
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**IRQACK_SECURE**	IRQACK_SECURE	**Interrupt acknowledgement for GPT7(secure)** 0: No effect 1: Associated interrupt request is acknowledged and should be relinquished.

100080a8 GPT8_CON **GPT8 Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											**MODE8**				**CLR8**	**EN8**
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE8**	MODE8	**Operation mode of GPT8** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR8**	CLR8	**Clears the counter of GPT8 to 0** 0: No effect 1: Clear
0	**EN8**	EN8	**Enables GPT8** 0: Disable 1: Enable

100080ac GPT8_CLK **GPT8 Clock Setting** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												**CLK8**	**CLKDIV8**			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK8**	CLK8	**Sets up clock source of GPT8** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV8**	CLKDIV8	**Setting of GPT8 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3

Bit(s)	Mnemonic	Name	Description
			0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

100080b0 GPT8_COUNT GPT8 Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER8[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER8[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER8**	COUNTER8	Timer counter of GPT8

100080b4 GPT8_COMPARE GPT8 Compare Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE8[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE8[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE8**	COMPARE8	Compare value of GPT8

100080b8 GPT9_CON GPT9 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE9				CLR9	EN9
Type											RW				WO	RW

100080b8 GPT9_CON **GPT9 Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reset												0	0			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset															0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE9**	MODE9	**Operation mode of GPT9** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR9**	CLR9	**Clears the counter of GPT9 to 0** 0: No effect 1: Clear
0	**EN9**	EN9	**Enables GPT9** 0: Disable 1: Enable

100080bc GPT9_CLK **GPT9 Clock Setting** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												CLK9	CLKDIV9			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK9**	CLK9	**Sets up clock source of GPT9** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV9**	CLKDIV9	**Setting of GPT9 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

100080c0 GPT9_COUNT **GPT9 Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER9[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER9[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER9**	COUNTER9	Timer counter of GPT9

100080c4 GPT9_COMPARE **GPT9 Compare Value** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE9[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE9[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE9**	COMPARE9	Compare value of GPT9

100080c8 GPT10_CON **GPT10 Control** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											MODE10				CLR10	EN10
Type											RW				WO	RW
Reset											0	0			0	0

Bit(s)	Mnemonic	Name	Description
5:4	**MODE9**	MODE9	**Operation mode of GPT9** 00: ONE-SHOT mode 01: REPEAT mode 10: KEEP-GO mode 11: FREERUN mode
1	**CLR9**	CLR9	**Clears the counter of GPT9 to 0** 0: No effect 1: Clear
0	**EN9**	EN9	**Enables GPT9** 0: Disable

Bit(s)	Mnemonic	Name	Description
			1: Enable

100080cc GPT10_CLK **GPT10 Clock Setting** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CLK10	CLKDIV10			
Type												RW	RW			
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4	**CLK10**	CLK10	**Sets up clock source of GPT10** 0: System clock (13MHz) 1: RTC clock (32kHz)
3:0	**CLKDIV10**	CLKDIV10	**Setting of GPT10 input clock frequency divider** 0000: Clock source divided by 1 0001: Clock source divided by 2 0010: Clock source divided by 3 0011: Clock source divided by 4 0100: Clock source divided by 5 0101: Clock source divided by 6 0110: Clock source divided by 7 0111: Clock source divided by 8 1000: Clock source divided by 9 1001: Clock source divided by 10 1010: Clock source divided by 11 1011: Clock source divided by 12 1100: Clock source divided by 13 1101: Clock source divided by 16 1110: Clock source divided by 32 1111: Clock source divided by 64

100080d0 GPT10_COUNT **GPT10 Counter** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COUNTER10[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COUNTER10[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COUNTER10**	COUNTER10	Timer counter of GPT10

100080d4 GPT10_COMPARE **GPT10 Compare Value** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COMPARE10[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COMPARE10[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**COMPARE10**	COMPARE10	Compare value of GPT10

8 UART

8.1 Introduction

The UARTs provide full duplex serial communication channels between the baseband chipset and external devices.

UART has both M16C450 and M16550A modes of operation, which are compatible with a range of standard software drivers. The extensions are designed to be broadly software compatible with 16550A variants, but certain areas offer no consensus.

In common with M16550A, the UART supports word lengths from 5 to 8 bits, an optional parity bit and one or two stop bits and is fully programmable by an 8-bit CPU interface. A 16-bit programmable baud rate generator and an 8-bit scratch register are included, together with separate transmit and receive FIFOs. 2 modem control lines and a diagnostic loop-back mode are provided. UART also includes two DMA handshake lines, indicating when the FIFOs is ready to transfer data to the CPU. Interrupts can be generated from any of the 10 sources.

Note that UART is designed so that all internal operation is synchronized by the CLK signal. This synchronization results in minor timing differences between the UART and industry standard 16550A device, which means that the core is not clock for clock identical to the original device.

After hardware reset, UART will be in M16C450 mode. Its FIFOs can then be enabled and UART can enter M16550A mode. UART has further additional functions beyond the M16550A mode. Each of the extended functions can be selected individually under software control.

UART provides more powerful enhancements than the industry-standard 16550:

Hardware flow control
This feature is very useful when the ISR latency is hard to predict and control in the embedded applications. The MCU is relieved of having to fetch the received data within a fixed amount of time.

Note that in order to enable the enhancements, and the enhanced mode bit, EFR[4], must be set. If EFR[4] is not set, IER[7:5], FCR[5:4], ISR[5:4] and MCR[7:6] cannot be written. The enhanced mode bit ensures that UART is backward compatible with the software that has been written for 16C450 and 16550A devices.

8.2 Feature list

- Provides 3 channels
- DMA, polling or interrupt operation
- Supports word lengths from 5 to 8 bits, with an optional parity bit and one or two stop bits

- UART0 for hardware automatic flow control
- Supports baud rates from 110bps up to 961,200bps
- Baud rate auto detection function

8.3 Block Diagram

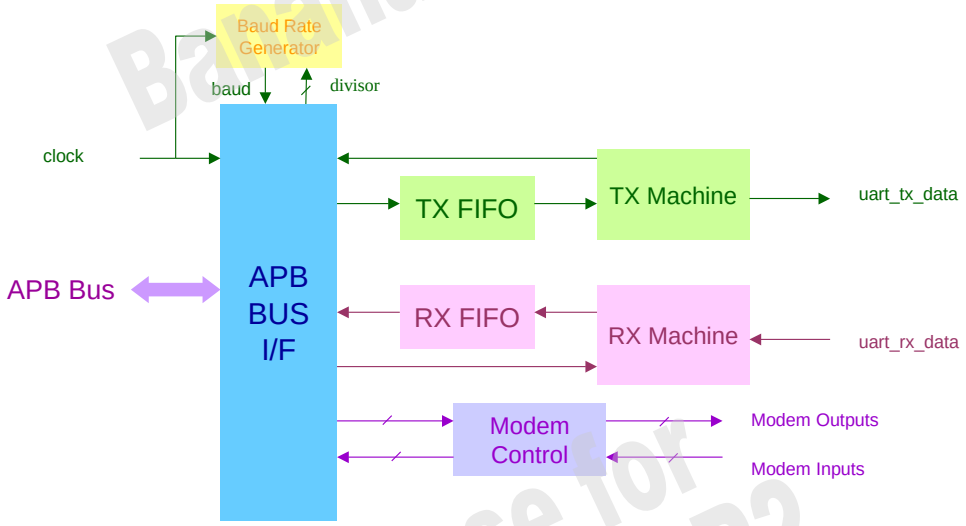


Figure 8-1: Block Diagram of UART

8.4 Register Definition

Table 8-1: UART Base Address Table

UART number	Base address	Feature
UART0	0x11002000	Supports DMA, HW flow control
UART1	0x11003000	Supports DMA, HW flow control
UART2	0x11004000	Supports DMA, HW flow control
UART3	0x11005000	Supports DMA, HW flow control

There are 4 UART IPs in this SOC. Use of the registers below are the same except that the base address needs to be changed to respective one.

Module name: UART base address: (+11002000h)

Address	Name	Width	Register function
11002000	**UARTn_RBR**	16	RX buffer register
11002000	**UARTn_THR**	16	TX holding register
11002000	**UARTn_DLL**	16	Divisor Latch (LS)
11002004	**UARTn_IER**	16	**Interrupt enable register** By storing 1 to a specific bit position, the interrupt associated with that bit will be enabled. Otherwise, the

Address	Name	Width	Register function
			interrupt will be disabled. IER[3:0] are modified when LCR[7] = 0. IER[7:4] are modified when LCR[7] = 0 & EFR[4] = 1.
11002004	**UARTn_DLM**	16	Divisor Latch (MS)
11002008	**UARTn_IIR**	16	**Interrupt identification register** Identifies if there are pending interrupts. ID4 and ID3 are presented only when EFR[4] = 1.
11002008	**UARTn_FCR**	16	**FIFO control register** FCR is used to control the trigger levels of the FIFOs or flush the FIFOs FCR[7:6] is modified when LCR != BFh. FCR[5:4] is modified when LCR != BFh & EFR[4] = 1. FCR[4:0] is modified when LCR != BFh.
11002008	**UARTn_EFR**	16	**Enhanced feature register** Note: Only when LCR=BF'h
1100200C	**UARTn_LCR**	16	**Line control register** Determines characteristics of serial communication signals. Modified when LCR[7] = 0.
11002010	**UARTn_MCR**	16	**Modem control register** Controls interface signals of the UART. MCR[5:0] are modified when LCR != 8'hBF. MCR[7] are modified when LCR != 8'hBF & EFR[4] = 1.
11002010	**UARTn_XON1**	16	**XON1** Note: XON1, XON2, XOFF1, XOFF2 are valid only when LCR=BF'h.
11002014	**UARTn_LSR**	16	**Line status register** Modified when LCR != BFh.
11002014	**UARTn_XON2**	16	XON2
11002018	**UARTn_MSR**	16	**Modem status register** Note: After a reset, D4-D7 are inputs. A modem status interrupt can be cleared by writing 0 or set by writing 1 to this register. D0-D3 can be written to. Modified when LCR[7] = 0.
11002018	**UARTn_XOFF1**	16	XOFF1
1100201C	**UARTn_SCR**	16	**Scratch register** A general purpose read/write register. After reset, its value is un-defined. Modified when LCR != BFh.
1100201C	**UARTn_XOFF2**	16	XOFF2
11002020	**UARTn_AUTOBAUD_EN**	16	AUTOBAUD_EN
11002024	**UARTn_HIGHSPEED**	16	HIGH SPEED UART

Address	Name	Width	Register function
11002028	**UARTn_SAMPLE_COUNT**	16	**SAMPLE_COUNT** When HIGHSPEED = 3, sample_count will be the threshold value for UART sample counter (sample_num). Counts from 0 to sample_count.
1100202C	**UARTn_SAMPLE_POINT**	16	**SAMPLE_POINT** When HIGHSPEED = 3, UART gets the input data when sample_count=sample_num, e.g. system clock = 13MHz, 921600 = 13000000/14 sample_count = 13 and sample point = 6 (sample the central point to decrease the inaccuracy) SAMPLE_POINT is usually (SAMPLE_COUNT-1)/2 without the decimal.
11002030	**UARTn_AUTOBAUD_REG**	16	AUTOBAUD_REG
1100203C	**UARTn_GUARD**	16	Guard time added register
11002040	**UARTn_ESCAPE_DAT**	16	Escape character register
11002044	**UARTn_ESCAPE_EN**	16	Escape enable register
11002048	**UARTn_SLEEP_EN**	16	Sleep enable register
1100204C	**UARTn_DMA_EN**	16	DMA enable register
11002050	**UARTn_RXTRI_AD**	16	Rx trigger address
11002054	**UARTn_FRACDIV_L**	16	Fractional divider LSB address
11002058	**UARTn_FRACDIV_M**	16	Fractional divider MSB address
1100205C	**UARTn_FCR_RD**	16	FIFO control register
11002060	**DEBUG0**	16	Debug register 0
11002064	**DEBUG1**	16	Debug register 1
11002090	**RX_SEL**	16	UART RX pin sel

11002000 UARTn_RBR RX Buffer Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RBR							
Type									RU							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**RBR**	RBR	**RX buffer register** Read-only register. The received data can be read by accessing this register. Modified when LCR[7] = 0.

11002000 UARTn_THR TX Holding Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									THR							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**THR**	THR	**TX holding register** Write-only register. The data to be transmitted are written to this register and sent to the PC via serial communication. Modified when LCR[7] = 0.

11002000 UARTn_DLL Divisor Latch (LS) 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DLL							
Type									RW							
Reset									0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
7:0	**DLL**	DLL	Modified when LCR[7] = 1.

11002004 UARTn_IER Interrupt Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CTSI	RTSI	XOFFI	RX_ABOVE_TRIG	EDSSI	ELSI	ETBEI	ERBFI
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Overview: By storing 1 to a specific bit position, the interrupt associated with that bit will be enabled. Otherwise, the interrupt will be disabled. IER[3:0] are modified when LCR[7] = 0. IER[7:4] are modified when LCR[7] = 0 & EFR[4] = 1.

Bit(s)	Mnemonic	Name	Description
7	**CTSI**	CTSI	**Masks an interrupt that is generated when a rising edge is detected on the CTS modem control line** *Note: This interrupt is only enabled when hardware flow control is enabled.* 0: Mask an interrupt that is generated when a rising edge is detected

Bit(s)	Mnemonic	Name	Description
			on the CTS modem control line. 1: Unmask an interrupt that is generated when a rising edge is detected on the CTS modem control line.
6	**RTSI**	RTSI	**Masks an interrupt that is generated when a rising edge is detected on the RTS modem control line** *Note: This interrupt is only enabled when hardware flow control is enabled.* 0: Mask an interrupt that is generated when a rising edge is detected on the RTS modem control line. 1: Unmask an interrupt that is generated when a rising edge is detected on the RTS modem control line.
5	**XOFFI**	XOFFI	**Masks an interrupt that is generated when an XOFF character is received** *Note: This interrupt is only enabled when software flow control is enabled.* 0: Mask an interrupt that is generated when an XOFF character is received. 1: Unmask an interrupt that is generated when an XOFF character is received.
4	**RX_ABOVE_TRIG**	RX_ABOVE_TRIG	**Masks Rx above trigger interrupt when rx_dma is enabled** 0: Mask Rx above trigger interrupt when rx_dma is enabled. 1: Unmask Rx above trigger interrupt when rx_dma is enabled.
3	**EDSSI**	EDSSI	**When set to 1, an interrupt will be generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set** 0: No interrupt is generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set. 1: An interrupt is generated if DDCD, TERI, DDSR or DCTS (MSR[4:1]) becomes set.
2	**ELSI**	ELSI	**When set to 1, an interrupt will be generated if BI, FE, PE or OE (LSR[4:1]) becomes set.** 0: No interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set. 1: An interrupt is generated if BI, FE, PE or OE (LSR[4:1]) becomes set.
1	**ETBEI**	ETBEI	**When set to 1, an interrupt will be generated if the TX holding register is empty or the contents of the TX FIFO are reduced to its trigger level** 0: No interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO are reduced to its trigger level. 1: An interrupt is generated if the TX Holding Register is empty or the contents of the TX FIFO are reduced to its trigger level
0	**ERBFI**	ERBFI	**When set to 1, an interrupt will be generated if the RX buffer contains data** 0: No interrupt is generated if the RX buffer contains data. 1: An interrupt is generated if the RX buffer contains data.

11002004 UARTn_DLM Divisor Latch (MS) 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DLM							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**DLM**	DLM	Modified when LCR[7] = 1

11002008 UARTn_IIR Interrupt Identification Register 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											**ID4**	**ID3**	**ID2**	**ID1**	**ID0**	**NINT**
Type											RU	RU	RU	RU	RU	RU
Reset											0	0	0	0	0	1

Overview: Identifies if there are pending interrupts. ID4 and ID3 are presented only when EFR[4] = 1.

Bit(s)	Mnemonic	Name	Description
5	**ID4**	ID4	
4	**ID3**	ID3	
3	**ID2**	ID2	
2	**ID1**	ID1	
1	**ID0**	ID0	
0	**NINT**	NINT	

11002008 UARTn_FCR FIFO Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**RFTL1_RFTL0**		**TFTL1_TFTL0**					**FIFOE**
Type									WO		WO					WO
Reset									0	0	0	0				0

Overview: FCR is used to control the trigger levels of the FIFOs or flush the FIFOs. FCR[7:6] is modified when LCR != BFh. FCR[5:4] is modified when LCR != BFh & EFR[4] = 1. FCR[4:0] is modified when LCR != BFh.

Bit(s)	Mnemonic	Name	Description
7:6	**RFTL1_RFTL0**	RFTL1_RFTL0	**RX FIFO trigger threshold** RX FIFO contains total 24 bytes. 0: 1 1: 6 2: 12 3: RXTRIG
5:4	**TFTL1_TFTL0**	TFTL1_TFTL0	**TX FIFO trigger threshold** TX FIFO contains total 16 bytes. 0: 1 1: 4 2: 8 3: 14 (FIFOSIZE - 2)
0	**FIFOE**	FIFOE	**Enables FIFO** This bit must be set to 1 for any of the other bits in the registers to have any effect.

Bit(s)	Mnemonic	Name	Description
			0: Disable both RX and TX FIFOs 1: Enable both RX and TX FIFOs.

11002008 UARTn_EFR Enhanced Feature Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									AUTO_CTS	AUTO_RTS			SW_FLOW_CONT			
Type									RW	RW			RW			
Reset									0	0			0	0	0	0

Overview: Only when LCR = BF'h

Bit(s)	Mnemonic	Name	Description
7	**AUTO_CTS**	AUTO_CTS	**Enables hardware transmission flow control** 0: Disable 1: Enable
6	**AUTO_RTS**	AUTO_RTS	**Enables hardware reception flow control** 0: Disable 1: Enable.
3:0	**SW_FLOW_CONT**	SW_FLOW_CONT	**Software flow control bits** 00xx: No TX flow control 10xx: Transmit XON1/XOFF1 as flow control bytes 01xx: Transmit XON2/XOFF2 as flow control bytes 11xx: Transmit XON1 & XON2 and XOFF1 & XOFF2 as flow control words xx00: No RX flow control xx10: Receive XON1/XOFF1 as flow control bytes xx01: Receive XON2/XOFF2 as flow control bytes xx11: Receive XON1 & XON2 and XOFF1 & XOFF2 as flow control words

1100200C UARTn_LCR Line Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DLAB	SB	SP	EPS	PEN	STB	WLS1_WLS0	
Type									RW	RW	RW	RW	RW	RW	RW	
Reset									0	0	0	0	0	0	0	0

Overview: Determines characteristics of serial communication signals. Modified when LCR[7] = 0.

Bit(s)	Mnemonic	Name	Description
7	**DLAB**	DLAB	**Divisor latch access bit** 0: The RX and TX registers are read/written at Address 0 and the IER register is read/written at Address 4. 1: The Divisor Latch LS is read/written at Address 0 and the Divisor Latch MS is read/written at Address 4.
6	**SB**	SB	**Sets up break**

Bit(s)	Mnemonic	Name	Description
			0: No effect 1: SOUT signal is forced into the "0" state.
5	**SP**	SP	**Stick parity** 0: No effect. 1: The parity bit is forced into a defined state, depending on the states of EPS and PEN: If EPS = 1 & PEN = 1, the parity bit is set and checked = 0. If EPS = 0 & PEN = 1, the parity bit is set and checked = 1.
4	**EPS**	EPS	**Selects even parity** 0: When EPS = 0, an odd number of ones is sent and checked. 1: When EPS = 1, an even number of ones is sent and checked.
3	**PEN**	PEN	**Enables parity** 0: The parity is neither transmitted nor checked. 1: The parity is transmitted and checked.
2	**STB**	STB	**Number of STOP bits** 0: One STOP bit is always added. 1: Two STOP bits are added after each character is sent; unless the character length is 5 when 1 STOP bit is added.
1:0	**WLS1_WLS0**	WLS1_WLS0	**Selects word length** 0: 5 bits 1: 6 bits 2: 7 bits 3: 8 bits

11002010 UARTn_MCR Modem Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									**XOFF_STATUS**			**Loop**	**OUT2**	**OUT1**	**RTS**	**DTR**
Type									RU			RW	RW	RW	RW	RW
Reset									0			0	0	0	0	0

Overview: Control interface signals of the UART. MCR[5:0] are modified when LCR != 8'hBF, MCR[7] are modified when LCR != 8'hBF & EFR[4] = 1.

Bit(s)	Mnemonic	Name	Description
7	**XOFF_STATUS**	XOFF_STATUS	**A read-only bit** 0: When an XON character is received 1: When an XOFF character is received
4	**Loop**	Loop	**Loop-back control bit.** 0: No loop-back is enabled. 1: Loop-back mode is enabled.
3	**OUT2**	OUT2	**Controls the state of the output NOUT2, even in loop mode.** 0: NOUT2 = 1 1: NOUT2 = 0
2	**OUT1**	OUT1	**Controls the state of the output NOUT1, even in loop mode.** 0: NOUT1 = 1 1: NOUT1 = 0
1	**RTS**	RTS	**Controls the state of the output NRTS, even in loop mode.**

Bit(s)	Mnemonic	Name	Description
			0: NRTS = 1 1: NRTS = 0
0	**DTR**	DTR	**Controls the state of the output NDTR, even in loop mode.** 0: NDTR = 1 1: NDTR = 0

11002010 UARTn_XON1 XON1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XON1							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Overview: XON1, XON2, XOFF1, XOFF2 are valid only when LCR=BF'h.

Bit(s)	Mnemonic	Name	Description
7:0	**XON1**	XON1	Auto-baud enabling signal

11002014 UARTn_LSR Line Status Register 0060

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									FIFOERR	TEMT	THRE	BI	FE	PE	OE	DR
Type									RU	RU	RU	RU	RU	RU	RU	RU
Reset									0	1	1	0	0	0	0	0

Overview: Modified when LCR != BFh.

Bit(s)	Mnemonic	Name	Description
7	**FIFOERR**	FIFOERR	**RX FIFO error indicator** 0: No PE, FE, BI set in the RX FIFO. 1: Set to 1 when there is at least one PE, FE or BI in the RX FIFO.
6	**TEMT**	TEMT	**TX holding register (or TX FIFO) and the TX shiftrRegister are empty** 0: Empty conditions below are not met. 1: If FIFOs are enabled, the bit is set whenever the TX FIFO and the TX shift register are empty. If FIFOs are disabled, the bit is set whenever TX holding register and TX shift register are empty.
5	**THRE**	THRE	**Indicates if there is room for TX holding register or TX FIFO is reduced to its trigger level** 0: Reset whenever the contents of the TX FIFO are more than its trigger level (FIFOs are enabled), or whenever TX holding register is not empty (FIFOs are disabled). 1: Set whenever the contents of the TX FIFO are reduced to its trigger level (FIFOs are enabled), or whenever TX holding register is empty and ready to accept new data (FIFOs are disabled).
4	**BI**	BI	**Break interrupt** 0: Reset by the CPU reading this register 1: If FIFOs are disabled, this bit is set whenever the SIN is held in

Bit(s)	Mnemonic	Name	Description
			the 0 state for more than one transmission time (START bit + DATA bits + PARITY + STOP bits). If the FIFOs are enabled, this error is associated with a corresponding character in FIFO and is flagged when this byte is at the top of FIFO. When a break occurs, only one zero character is loaded into FIFO: the next character transfer is enabled when SIN enters the marking state and receives the next valid start bit.
3	**FE**	FE	**Framing error** 0: Reset by the CPU reading this register 1: If FIFOs are disabled, this bit will be set if the received data do not have a valid STOP bit. If FIFOs are enabled, the state of this bit will be revealed when the byte it refers to is the next to be read.
2	**PE**	PE	**Parity error** 0: Reset by the CPU reading this register 1: If FIFOs are disabled, this bit will be set if the received data do not have a valid parity bit. If FIFOs are enabled, the state of this bit is revealed when the referred byte is the next to be read.
1	**OE**	OE	**Overrun error** 0: Reset by the CPU reading this register. 1: If FIFOs are disabled, this bit will be set if the RX buffer is not read by the CPU before the new data from the RX shift register overwrites the previous contents. If FIFOs are enabled, an overrun error occurs when RX FIFO is full and the RX shift register becomes full. OE is set as soon as this happens. The character in the shift register is then overwritten, but not transferred to FIFO.
0	**DR**	DR	**Data ready** 0: Cleared by the CPU reading the RX buffer or by reading all the FIFO bytes. 1: Set by the RX buffer becoming full or by a byte being transferred into the FIFO.

11002014 UARTn_XON2 XON2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XON2							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**XON2**	XON2	

11002018 UARTn_MSR Modem Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												CTS				DCTS
Type												RO				RW
Reset												0				0

Overview: After a reset, D4-D7 are inputs. A modem status interrupt can be cleared by writing 0 or set by writing 1 to this register. D0-D3 can be written to. Modified when LCR[7] = 0.

Bit(s)	Mnemonic	Name	Description
4	**CTS**	CTS	**Clear to send** When Loop = 0, this value is the complement of the NCTS input signal. When Loop = 1, this value is equal to the RTS bit in the modem control register.
0	**DCTS**	DCTS	**Delta clear to send** 0: Cleared if the state of CTS has not changed since this register was last read. 1: Set if the state of CTS has changed since this register was last read.

11002018 UARTn_XOFF1 XOFF1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XOFF1							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**XOFF1**	XOFF1	

1100201C UARTn_SCR Scratch Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SCR							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Overview: A general-purpose read/write register. After reset, its value will be un-defined. Modified when LCR != BFh.

Bit(s)	Mnemonic	Name	Description
7:0	**SCR**	SCR	

UARTn_XOFF2 XOFF2 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XOFF2							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**XOFF2**	XOFF2	

11002020 UARTn_AUTOBAUD_EN AUTOBAUD_EN 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															AUTOBAUD_SEL	AUTO_EN
Type															RW	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1	**AUTOBAUD_SEL**	AUTOBAUD_SEL	**Selects auto-baud** 0: No effect 1: Does not fix baud rate at standard value
0	**AUTO_EN**	AUTOBAUD_EN	**Auto-baud enabling signal** 0: Disable auto-baud function 1: Enable auto-baud function (UARTn+0024h SPEED should be set 0.)

11002024 UARTn_HIGHSPEED HIGH SPEED UART 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															SPEED	
Type															RW	
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1:0	**SPEED**	SPEED	**UART sample counter base** 0: Based on 16*baud_pulse, baud_rate = system clock frequency/16/{DLH, DLL} 1: Based on 8*baud_pulse, baud_rate = system clock frequency/8/{DLH, DLL} 2: Based on 4*baud_pulse, baud_rate = system clock frequency/4/{DLH, DLL} 3: Based on sampe_count * baud_pulse, baud_rate = system clock frequency/sampe_count/{DLM, DLL}

11002028 UARTn_SAMPLE_COUNT SAMPLE_COUNT 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SAMPLECOUNT							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Overview: When HIGHSPEED = 3, sample_count will be the threshold value for UART sample counter (sample_num). Counts from 0 to sample_count.

Bit(s)	Mnemonic	Name	Description
7:0	**SAMPLECOUNT**	SAMPLECOUNT	

1100202C UARTn_SAMPLE_POINT SAMPLE_POINT 00FF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SAMPLEPOINT							
Type									RW							
Reset									1	1	1	1	1	1	1	1

Overview: When HIGHSPEED = 3, UART gets the input data when sample_count=sample_num, e.g. system clock = 13MHz, 921600 = 13000000/14. sample_count = 13 and sample point = 6 (sample the central point to decrease the inaccuracy). SAMPLE_POINT is usually (SAMPLE_COUNT-1)/2 without the decimal.

Bit(s)	Mnemonic	Name	Description
7:0	**SAMPLEPOINT**	SAMPLEPOINT	

11002030 UARTn_AUTOBAUD_REG AUTOBAUD_REG 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									BAUD_STAT				BAUD_RATE			
Type									RO				RO			
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4	**BAUD_STAT**	BAUD_STAT	**Autobaud format** 0: Autobaud is detecting. 1: AT_7N1 2: AT_7O1 3: AT_7E1 4: AT_8N1 5: AT_8O1 6: AT_8E1 7: at_7N1 8: at_7E1 9: at_7O1 10: at_8N1 11: at_8E1 12: at_8O1 13: Autobaud detection fails.
3:0	**BAUD_RATE**	BAUD_RATE	**Autobaud baud rate** 0: 115,200 1: 57,600 2: 38,400 3: 19,200 4: 9,600 5: 4,800 6: 2,400 7: 1,200 8: 300 9: 110

1100203C UARTn_GUARD Guard time added register 000F

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name												GUARD_EN	GUARD_CNT			
Type												RW	RW			
Reset												0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
4	**GUARD_EN**	GUARD_EN	**Guard interval added enabling signal** 0: No guard interval added 1: Add guard interval after stop bit
3:0	**GUARD_CNT**	GUARD_CNT	**Guard interval count value** Guard interval = [1/(system clock/div_step/div)]*GUARD_CNT

11002040 UARTn_ESCAPE_DAT Escape character register 00FF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									ESCAPE_DAT							
Type									RW							
Reset									1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
7:0	**ESCAPE_DAT**	ESCAPE_DAT	**Escapes character added before software flow control data and escape character** For example, if Tx data are xon (31h), with esc_en =1, UART transmits data as esc + CEh (~xon).

11002044 UARTn_ESCAPE_EN Escape enable register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ESC_EN
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**ESC_EN**	ESC_EN	**Adds escape character in transmitter and removes escape character in receiver by UART** 0: Does not deal with the escape character 1: Add escape character in transmitter and remove escape character in receiver.

11002048 UARTn_SLEEP_EN Sleep enable register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SLEEP_EN
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
0	**SLEEP_EN**	SLEEP_EN	**For sleep mode issue** 0: Does not deal with sleep mode indication signal 1: Activate hardware flow control or software control according to software initial settings when the chip enters the sleep mode. Release the hardware flow when the chip wakes up. However, for software control, UART sends XON when being awaken and when FIFO does not reach the threshold level.

1100204C UARTn_DMA_EN DMA enable register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														TO_CNT_AUTORST	TX_DMA_EN	RX_DMA_EN
Type														RW	RW	RW
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2	**TO_CNT_AUTORST**	TO_CNT_AUTORST	**Time-out counter auto reset register** 0: After RX time-out happens, SW shall reset the interrupt by reading UART 0x4C. 1: The time-out counter will be auto reset. Set this register when Rain's new DMA is used.
1	**TX_DMA_EN**	TX_DMA_EN	**TX_DMA mechanism enabling signal** 0: Does not use DMA in TX 1: Use DMA in TX. When this register is enabled, the flow control is based on the DMA threshold and generates a time-out interrupt for DMA.
0	**RX_DMA_EN**	RX_DMA_EN	**RX_DMA mechanism enabling signal** 0: Does not use DMA in RX 1: Use DMA in RX. When this register is enabled, the flow control is based on the DMA threshold and generates a time-out interrupt

11002050 UARTn_RXTRI_AD Rx Trigger Address 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													RXTRIG			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**RXTRIG**	RXTRIG	**When {rtm,rtl}=2'b11, Rx FIFO threshold will be Rxtrig.** The value is suggested to be smaller than half the RX FIFO size, which is 24 bytes.

11002054 UARTn_FRACDIV_L Fractional Divider LSB Address 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name									FRACDIV_L							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**FRACDIV_L**	FRACDIV_L	Adds sampling count (+1) from state data7 to data0 to contribute to fractional divisor

11002058 UARTn_FRACDIV_M Fractional Divider MSB Address 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															FRACDIV_M	
Type															RW	
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1:0	**FRACDIV_M**	FRACDIV_M	Adds sampling count when in state stop to parity to contribute to fractional divisor

1100205C UARTn_FCR_RD FIFO Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RFTL1_RFTL0		TFTL1_TFTL0					FIFOE
Type									RO		RO					RO
Reset									0	0	0	0				0

Bit(s)	Mnemonic	Name	Description
7:6	**RFTL1_RFTL0**	RFTL1_RFTL0	**RX FIFO trigger threshold** RX FIFO contains total 24 bytes. 0: 1 1: 6 2: 12 3: RXTRIG
5:4	**TFTL1_TFTL0**	TFTL1_TFTL0	**TX FIFO trigger threshold** TX FIFO contains total 16 bytes. 0: 1 1: 4 2: 8 3: 14 (FIFOSIZE - 2)
0	**FIFOE**	FIFOE	**Enables FIFO** This bit must be set to 1 for any of the other bits in the registers to have any effect. 0: Disable both RX and TX FIFOs 1: Enable both RX and TX FIFOs

11002060 DEBUG0 Debug Register 0 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name												DEBUG0				
Type												RU				
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:0	**DEBUG0**	DEBUG0	TX state machine

11002064 DEBUG1 Debug Register 1 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													DEBUG1			
Type													RU			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**DEBUG1**	DEBUG1	RX state machine

11002090 RX_SEL UART RX Pin Sel 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																RX_SEL
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**RX_SEL**	RX_SEL	**Selects RX pin** *Note: Only UART 0 can choose USB pin*. 0: Choose UART RX pin 1: Choose USB pin

9 I2C

9.1 Introduction

I2C (Inter-IC)/SCCB (Serial Camera Control Bus) is a two-wire serial interface. The two signals are SCL and SDA. SCL is a clock signal that is driven by the master. SDA is a bi-directional data signal that can be driven by either the master or the slave. This generic controller supports the master role and conforms to the I2C specification.

9.2 Feature list

- I2C compliant master mode operation
- Adjustable clock speed for LS/FS mode operation
- Supports 7-bit/10-bit addressing
- Supports high-speed mode
- Supports slave clock extension
- START/STOP/REPEATED START condition
- Manual transfer mode
- Multi-write per transfer
- Multi-read per transfer
- Multi-transfer per transaction
- Combined format transfer with length change capability.-
- Active drive/wired-and I/O configuration

9.2.1 Manual Transfer Mode

The controller offers manual mode

When the manual mode is selected, in addition to the slave address register, the controller has a built-in 8byte deep FIFO which allows MCU to prepare up to 8 bytes of data for a write transfer, or read up to 8 bytes of data for a read transfer.

9.3 Block Diagram

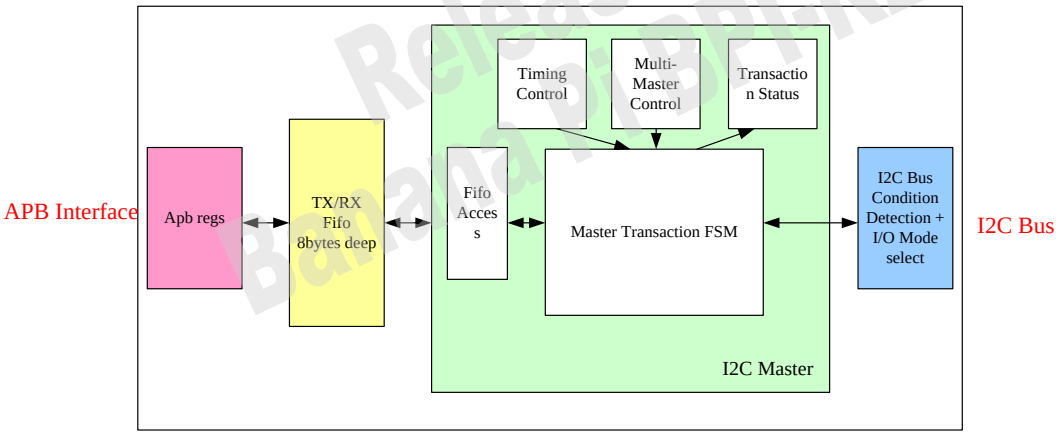


Figure 9-1: Block Diagram of I2C

9.4 Register Definition

Module name: I2C Base address: (+11007000h)

Address	Name	Width	Register function
11007000	**DATA_PORT**	16	Data port register
11007004	**SLAVE_ADDR**	16	Slave address register
11007008	**INTR_MASK**	16	**Interrupt mask register** This register provides masks for the corresponding interrupt sources as indicated in intr_stat register. 1 = Allow interrupt 0 = Disable interrupt *Note: While disabled, the corresponding interrupt will not be asserted, however the intr_stat will still be updated with the status, i.e. mask does not affect intr_stat register values.*
1100700C	**INTR_STAT**	16	**Interrupt status register** When an interrupt is issued by I2C controller, this register will need to be read by MCU to determine the cause for the interrupt. After this status has been read and appropriate actions are taken, the corresponding interrupt source will need to be written 1 to clear.
11007010	**CONTROL**	16	Control register
11007014	**TRANSFER_LEN**	16	Transfer length register (number of bytes per transfer)
11007018	**TRANSAC_LEN**	16	Transaction length register (number of transfers per transaction)
1100701C	**DELAY_LEN**	16	Inter delay length register
11007020	**TIMING**	16	**Timing controlregister** LS/FS only. This register is used to control the output waveform timing. Each half pulse width, i.e. each high or low pulse, is equal to

Address	Name	Width	Register function
			step_cnt_div+1*(sample_cnt_div+1)*1/4*BASE_CLOCK_PERIOD
11007024	**START**	16	Start register
11007028	**EXT_CONF**	16	Extension configuration register
11007030	**FIFO_STAT**	16	FIFO status register
11007034	**FIFO_THRESH**	16	**FIFO thresh register** (For debugging only) By default, these values do not need to be adjusted. *Note: For RX, no time-out mechanism is implemented. Therefore, RX_trig_thresh must be left at 0, or there would be data left in FIFO that is not fetched by the DMA controller.*
11007038	**FIFO_ADDR_CLR**	16	FIFO address clear register
11007040	**IO_CONFIG**	16	**IO config register** This register is used to configure the I/O for the SDA and SCL lines to select between normal I/O mode, or open-drain mode to support wired-and bus.
11007044	**MULTIMAS**	16	**Multiple I2C masters** This registers contains options for supporting multi-master features.
11007048	**HS**	16	**High speed mode register** This register contains options for supporting high speed operation features. Each HS half pulse width, i.e. each high or low pulse, is equal to step_cnt_div*(sample_cnt_div*1/16.25MHz)
11007050	**SOFTRESET**	16	Soft reset register
11007054	**HW_DCM_EN**	16	HW DCM enable
11007064	**DEBUGSTAT**	16	Debug status register
11007068	**DEBUGCTRL**	16	Debug control register

11007000 DATA_PORT Data Port Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DATA_PORT							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**DATA_PORT**	DATA_PORT	**FIFO access port** During master write sequences (slave_addr[0] = 0), this port can be written by APB, and during master read sequences (slave_addr[0] = 1), this port can be read by APB. *Note: Slave_addr must be set correctly before accessing FIFO.* For debugging only: If the fifo_apb_debug bit is set, FIFO can be read and written by the APB.

11007004 SLAVE_ADDR Slave Address Register 00BF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SLAVE_ADDR							
Type									RW							
Reset									1	0	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
7:0	**SLAVE_ADDR**	SLAVE_ADDR	**Specifies the slave address of the device to be accessed** Bit 0 is defined by the I2C protocol as a bit that indicates the direction of transfer. 0: Master write 1: Master read

11007008 INTR_MASK Interrupt Mask Register 00FF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SPARE				MASK_ARB_LOST	MASK_HS_NACKERR	MASK_ACKERR	MASK_TRANSAC_COMP
Type									RW				RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1

Overview: This register provides masks for the corresponding interrupt sources as indicated in intr_stat register. 1 = allow interrupt; 0 = disable interrupt. Note that while disabled, the corresponding interrupt will not be asserted. However, intr_stat will still be updated with the status, i.e. mask does not affect intr_stat register values.

Bit(s)	Mnemonic	Name	Description
7:4	**SPARE**	SPARE	Reserved
3	**MASK_ARB_LOST**	MASK_ARB_LOST	Setting this value to 0 will mask ARB_LOST interrupt signal.
2	**MASK_HS_NACKERR**	MASK_HS_NACKERR	Setting this value to 0 will mask HS_NACKERR interrupt signal.
1	**MASK_ACKERR**	MASK_ACKERR	Setting this value to 0 will mask ACK_ERR interrupt signal.
0	**MASK_TRANSAC_COMP**	MASK_TRANSAC_COMP	Setting this value to 0 will mask TRANSAC_COMP interrupt signal.

1100700C INTR_STAT Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ARB_LOST	HS_NACKERR	ACKERR	TRANSAC_COMP
Type													W1	W1	W1	W1

1100700C INTR_STAT Interrupt Status Register 0000

													C	C	C	C
Reset													0	0	0	0

Overview: When an interrupt is issued by the I2C controller, this register will need to be read by MCU to determine the cause for the interrupt. After this status has been read and appropriate actions are taken, the corresponding interrupt source will need to be written 1 to clear.

Bit(s)	Mnemonic	Name	Description
3	**ARB_LOST**	ARB_LOST	This status is asserted if the I2C controller loses arbitration.
2	**HS_NACKERR**	HS_NACKERR	This status is asserted ifHS master code NACK error detection is enabled. If enabled, HS master code NACK err will cause transaction to end, and stop will be issued.
1	**ACKERR**	ACKERR	This status is asserted if ACK error detection is enabled. If enabled, ACKERR will cause transaction to end, and stop will be issued.
0	**TRANSAC_COMP**	TRANSAC_COMP	This status is asserted when a transaction is completed successfully.

11007010 CONTROL Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										**TRANSFER_LEN_CHANGE**	**ACKERR_DET_EN**	**DIR_CHANGE**	**CLK_EXT_EN**	**DMA_EN**	**RS_STOP**	
Type										RW	RW	RW	RW	RW	RW	
Reset										0	0	0	0	0	0	

Bit(s)	Mnemonic	Name	Description
6	**TRANSFER_LEN_CHANGE**	TRANSFER_LEN_CHANGE	**Specifies whether or not to change the transfer length after the fist transfer is completed** If enabled, the transfers after the first transfer will use the transfer_len_aux parameter.
5	**ACKERR_DET_EN**	ACKERR_DET_EN	**Enables slave ACK error detection** When enabled, if slave ACK error is detected, the master shall terminate the transaction by issuing a STOP condition and then asserts the ACKERR interrupt. MCU handles this case appropriately and then resets the FIFO address before reissuing transaction again. If this option is disabled, the controller will ignore slave ACK error and keep on scheduled transaction. 0: Disable 1: Enable
4	**DIR_CHANGE**	DIR_CHANGE	**Combined transfer format, where the direction of transfer is to be changed from write to read after the FIRST RS condition** *Note: When set to 1, the transfers after the direction change will be based on the transfer_len_aux parameter.* 0: Disable 1: Enable
3	**CLK_EXT_EN**	CLK_EXT_EN	**I2C spec allows slaves to hold the SCL line low if it is not yet ready for further processing.**

Bit(s)	Mnemonic	Name	Description
			Therefore, if this bit is set to 1, the master controller will enter a high wait state until the slave releases the SCL line.
2	**DMA_EN**	DMA_EN	**By default, this is disabled, and the FIFO data shall be manually prepared by MCU.** This default setting should be used for transfer sizes of smaller than 8 data bytes and no multiple transfer is configured. When enabled, DMA requests are turned on, and the FIFO data should be prepared in memory.
1	**RS_STOP**	RS_STOP	**In LS/FS mode, this bit affects multi-transfer transaction only.** It controls whether or not the REPEATED-START condition is used between transfers. The last ending transfer always ends with a STOP. In HS mode, this bit must be set to 1. 0: Use STOP 1: Use REPEATED-START

11007014 **TRANSFER_LEN** **Transfer Length Register (Number of Bytes per Transfer)** **0101**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				TRANSFER_LEN_AUX					TRANSFER_LEN							
Type				RW					RW							
Reset				0	0	0	0	1	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
12:8	**TRANSFER_LEN_AUX**	TRANSFER_LEN_AUX	**This field is valid only when dir_change is set to 1.** Indicates the number of data bytes to be transferred in 1 transfer unit (excluding slave address byte) for the transfers following the direction change. That is, if dir_change =1, then the first write transfer length will depend on transfer_len, while the second read transfer length depend on transfer_len_aux. Dir change is always after the first transfer. *Note: The value must be set to be bigger than 1; otherwise no transfer will take place.*
7:0	**TRANSFER_LEN**	TRANSFER_LEN	**Indicates the number of data bytes to be transferred in 1 transfer unit (excluding slave address byte)** *Note: The value must be set to be bigger than 1; otherwise no transfer will take place.*

11007018 **TRANSAC_LEN** **Transaction Length Register (Number of Transfers per Transaction)** **0001**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									TRANSAC_LEN							
Type									RW							
Reset									0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
7:0	**TRANSAC_LEN**	TRANSAC_LEN	**Indicates the number of transfers to be transferred in 1 transaction** *Note: The value must be set to be bigger than 1; otherwise no*

Bit(s)	Mnemonic	Name	Description
			transfer will take place.

1100701C DELAY_LEN Inter Delay Length Register 0002

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DELAY_LEN							
Type									RW							
Reset									0	0	0	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
7:0	DELAY_LEN	DELAY_LEN	**Sets up wait delay between consecutive transfers when RS_STOP bit is set to 0** Unit: Half the pulse width

11007020 TIMING Timing Control Register 1303

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DATA_READ_ADJ	DATA_READ_TIME				SAMPLE_CNT_DIV					STEP_CNT_DIV					
Type	RW	RW				RW					RW					
Reset	0	0	0	1		0	1	1			0	0	0	0	1	1

Overview: LS/FS only. This register is used to control the output waveform timing. Each half pulse width, i.e. each high or low pulse, is equal to step_cnt_div+1*(sample_cnt_div+1)*1/4*BASE_CLOCK_PERIOD.

Bit(s)	Mnemonic	Name	Description
15	DATA_READ_ADJ	DATA_READ_ADJ	When set to 1, data latch in sampling time during master reads are adjusted according to the DATA_READ_TIME value. Otherwise, by default, the data are latched in at half of the high pulse width point. This value must be set to be smaller than or equal to half the high pulse width.
14:12	DATA_READ_TIME	DATA_READ_TIME	This value is valid only when DATA_READ_ADJ is set to 1. This can be used to adjust so that the data are latched in at earlier sampling points (assuming data are settled by then)
10:8	SAMPLE_CNT_DIV	SAMPLE_CNT_DIV	Used for LS/FS only. This adjusts the width of each sample. (sample width = sample_cnt_div*1/4 BASE_CLOCK_PERIOD)
5:0	STEP_CNT_DIV	STEP_CNT_DIV	Specifies the number of samples per half pulse width, i.e. each high or low pulse

11007024 START Start Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																START
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**START**	START	**Starts the transaction on the bus** It is auto de-asserted at the end of the transaction.

11007028 EXT_CONF Extension Configuration Register 1800

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXT_TIME															EXT_EN
Type	RW															RW
Reset	0	0	0	1	1	0	0	0								0

Bit(s)	Mnemonic	Name	Description
15:8	**EXT_TIME**	EXT_TIME	**Configurable extension time of start condition** Time unit: 1/4 BASE_CLOCK_PERIOD *Note: The max. value is {0xFF - SAMPLE_CNT_DIV}.*
0	**EXT_EN**	EXT_EN	**Used for standard mode only (baud rate is up to 100kHz)** This option decides to perform the extension of start /stop condition. If enabled, perform the extension; otherwise not. 0: Disable 1: Enable

11007030 FIFO_STAT FIFO Status Register 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RD_ADDR				WR_ADDR				FIFO_OFFSET						WR_FULL	RD_EMPTY
Type	RU				RU				RU						RU	RU
Reset	0	0	0	0	0	0	0	0	0	0	0	0			0	1

Bit(s)	Mnemonic	Name	Description
15:12	**RD_ADDR**	RD_ADDR	**Current RD address pointer** Only bit [2:0] has physical meaning.
11:8	**WR_ADDR**	WR_ADDR	**Current WR address pointer** Only bit [2:0] has physical meaning.
7:4	**FIFO_OFFSET**	FIFO_OFFSET	**wr_addr[3:0] - rd_addr[3:0]**
1	**WR_FULL**	WR_FULL	**Indicates FIFO is full**
0	**RD_EMPTY**	RD_EMPTY	**Indicates FIFO is empty**

11007034 FIFO_THRESH FIFO Thresh Register 0700

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						TX_TRIG_THRESH								RX_TRIG_THRESH		
Type						RW								RW		
Reset						1	1	1						0	0	0

Overview: For debugging only. By default, these values do not need to be adjusted. Note that for RX, no time-out mechanism is implemented. Therefore, RX_trig_thresh must be left at 0, or there will be data left in FIFO that is not fetched by the DMA controller.

Bit(s)	Mnemonic	Name	Description
10:8	**TX_TRIG_THRESH**	TX_TRIG_THRESH	When Tx FIFO level is below this value, Tx DMA request is asserted.
2:0	**RX_TRIG_THRESH**	RX_TRIG_THRESH	When Rx FIFO level is above this value, Rx DMA request is asserted.

11007038 FIFO_ADDR_CLR FIFO Address Clear Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																FIFO_ADDR_CLR
Type																WO
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**FIFO_ADDR_CLR**	FIFO_ADDR_CLR	When written with 1'b1, a 1 pulse fifo_addr_clr is generated to clear the FIFO address to back to 0.

11007040 IO_CONFIG IO Config Register 0003

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													IDLE_OE_EN	IO_SYNC_EN	SDA_IO_CONFIG	SCL_IO_CONFIG
Type													RW	RW	RW	RW
Reset													0	0	1	1

Overview: This register is used to configure the I/O for the SDA and SCL lines to select between normal I/O mode, or open-drain mode to support wired-and bus.

Bit(s)	Mnemonic	Name	Description
3	**IDLE_OE_EN**	IDLE_OE_EN	0: Does not drive bus in idle state 1: Drive bus in idle state
2	**IO_SYNC_EN**	IO_SYNC_EN	**For debugging only** When set to 1, SCL and SDA inputs will be first dual synced by bclk_ck. This should not be needed. Only reserved for debugging.
1	**SDA_IO_CONFIG**	SDA_IO_CONFIG	0: Normal tristate I/O mode 1: Open-drain mode
0	**SCL_IO_CONFIG**	SCL_IO_CONFIG	0: Normal tristate I/O mode 1: Open-drain mode

11007044 MULTIMAS Multiple I2C Masters 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														BUS_DET_EN	CLK_SYNC_EN	ARB_EN
Type														RW	RW	RW
Reset														0	0	0

Overview: This registers contains options for supporting multi-master features.

Bit(s)	Mnemonic	Name	Description
2	**BUS_DET_EN**	BUS_DET_EN	When enabled, the bus status is monitored, and if the bus is currently busy, no new transaction can proceed.
1	**CLK_SYNC_EN**	CLK_SYNC_EN	When enabled, clk synchronization will be performed.
0	**ARB_EN**	ARB_EN	When enabled, multi-master arbitration will be performed.

11007048 HS High Speed Mode Register 0102

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		HS_SAMPLE_CNT_DIV				HS_STEP_CNT_DIV				MASTER_CODE					HS_NACKERR_DET_EN	HS_EN
Type		RW				RW				RW					RW	RW
Reset		0	0	0		0	0	1		0	0	0			1	0

Overview: This register contains options for supporting high speed operation features. Each HS half-pulse width, i.e. each high or low pulse, is equal to step_cnt_div*(sample_cnt_div*1/16.25MHz).

Bit(s)	Mnemonic	Name	Description
14:12	**HS_SAMPLE_CNT_DIV**	HS_SAMPLE_CNT_DIV	When the high-speed mode is entered after the master code transfer is completed, the sample width will become dependent on this parameter.
10:8	**HS_STEP_CNT_DIV**	HS_STEP_CNT_DIV	When the high-speed mode is entered after the master code transfer is completed, the number of samples per half pulse width will become dependent on this value.
6:4	**MASTER_CODE**	MASTER_CODE	This is the 3-bit programmable value for the master code to be transmitted.
1	**HS_NACKERR_DET_EN**	HS_NACKERR_DET_EN	**Enables NACKERR detection during the master code transmission** When enabled, if NACK is not received after the master code is transmitted, the transaction will be terminated with a STOP condition.
0	**HS_EN**	HS_EN	**Enables high-speed transaction** *Note: rs_stop must be set to 1.*

11007050 SOFTRESET Soft Reset Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

11007050 SOFTRESET Soft Reset Register 0000

Name																SOFT_RESET
Type																WO
Reset																0

Bit(s)	Mnemonic	Name	Description
0	SOFT_RESET	SOFT_RESET	When written with 1'b1, a 1 pulse soft reset is used as synchronous reset to reset the I2C internal hardware circuits.

11007054 HW_DCM_EN HW DCM Enable 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DCM_EN
Type																RW
Reset																1

Bit(s)	Mnemonic	Name	Description
0	DCM_EN	DCM_EN	**Enables HW DCM function** Default is enable. 0: Disable 1: Enable

11007064 DEBUGSTAT Debug Status Register 0020

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									BUS_BUSY	MASTER_WRITE	MASTER_READ	MASTER_STATE				
Type									RU	RU	RU	RU				
Reset									0	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	BUS_BUSY	BUS_BUSY	**For debugging only** Valid when bus_detect_en is 1. bus_busy = 1 indicates a start transaction has been detected and no stop condition has been detected yet.
6	MASTER_WRITE	MASTER_WRITE	**For debugging only** 1: Current transfer is in the master write dir
5	MASTER_READ	MASTER_READ	**For debugging only** 1: Current transfer is in the master read dir
4:0	MASTER_STATE	MASTER_STATE	**For debugging only. Reads back the current master_state.** 0: Idle state 1: I2C master is preparing sending out the start bit, SCL=1, SDA=1. 2: I2C master is sending out the start bit, SCL=1, SDA=0. 3: I2C master/slave is preparing transmitting data bit, SCL=0, DA=data bit (data bit can be changed when SCL=0). 4: I2C master/slave is transmitting data bit, SCL=1, SDA=data bit

Bit(s)	Mnemonic	Name	Description
			(data bit is stable when SCL=1). 5: I2C master/slave is preparing transmitting the ACK bit, SCL=0, SDA=ack (The ACK bit can be changed when SCL=0) 6: I2C master/slave is transmitting the ACK bit, SCL=1, SDA=0 (ack bit is stable when SCL=1) 7: I2C master is preparing sending out stop bit or repeated-start bit, SCL=0, SDA=0/1 (0: Stop bit; 1: Repeated-start bit). 8: I2C master is sending out stop bit or repeated-start bit, SCL=1, SDA=1/0 (1: Stop bit; 0: Repeated-start bit). 9: I2C master is in delay start between two transfers, SCL=1, SDA=1. 10: I2C master is in FIFO wait state; For writing transaction, it means FIFO is empty and I2Cc master is waiting for DMA controller writing data into FIFO; For reading transaction, it means FIFOo is full and I2C master is waiting for DMA controller reading data from FIFO, SCL=0, SDA=don't care. 12: I2C master is preparing sending out data bit of master code. This state is used only in high-speed transaction, SCL=0, SDA=data bit of master code (data bit of master code can be changed when SCL=0). 13: I2C master is sending out data bit of master code. This state is used only in high-speed transaction, SCL=1, SDA=data bit of master code (data bit of master code is stable when SCL=1) 14: I2C master/slave is preparing transmitting the NACK bit, SCL=0, SDA=nack bit (The NACK bit can be changed when SCL=0). This state is used only in high-speed transaction. 15: I2C master/slave is transmitting the NACK bit, SCL=1, SDA=1; This state is used only in high-speed transaction

11007068 DEBUGCTRL Debug Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															APB_DEBUG_RD	FIFO_APB_DEBUG
Type															WO	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1	**APB_DEBUG_RD**	APB_DEBUG_RD	**Only valid when fifo_apb_debug is set to 1** Writing to this register will generate a 1 pulsed FIFO APB RD signal for reading the FIFO data.
0	**FIFO_APB_DEBUG**	FIFO_APB_DEBUG	**Used for trace 32 debugging purposes** When using trace 32, and the memory map is shown, turning this bit on will block the normal APB read access. The APB read access to the FIFO is then enabled by writing to apb_debug_rd. 0: Disable 1: Enable

Module name: I2C_SCCB_Controller base address : (+11007000h)

Address	Name	Width	Register function

Address	Name	Width	Register function
11007000	**DATA_PORT**	16	Data port register
11007004	**SLAVE_ADDR**	16	Slave address register
11007008	**INTR_MASK**	16	**Interrupt mask register** This register provides masks for the corresponding interrupt sources as indicated in the intr_stat register. 1 = Allow interrupt 0 = Disable interrupt *Note: While disabled, the corresponding interrupt will not be asserted. However the intr_stat will still be updated with the status, i.e. mask does not affect intr_stat register values.*
1100700C	**INTR_STAT**	16	**Interrupt status register** When an interrupt is issued by I2C controller, this register will need to be read by MCU to determine the cause for the interrupt. After this status has been read and appropriate actions are taken, the corresponding interrupt source will need to be written 1 to clear.
11007010	**CONTROL**	16	Control register
11007014	**TRANSFER_LEN**	16	Transfer length register (number of bytes per transfer)
11007018	**TRANSAC_LEN**	16	Transaction length register (number of transfers per transaction)
1100701C	**DELAY_LEN**	16	Inter delay length register
11007020	**TIMING**	16	**Timing control register** LS/FS only. This register is used to control the output waveform timing. Each half pulse width, i.e. each high or low pulse, is equal to (step_cnt_div+1)*(sample_cnt_div+1)*1/4*BASE_CLOCK_PERIOD
11007024	**START**	16	Start register
11007028	**EXT_CONF**	16	Extension configuration register
11007030	**FIFO_STAT**	16	FIFO status register
11007034	**FIFO_THRESH**	16	**FIFO thresh register** For debugging only. By default, these values do not need to be adjusted. *Note: For RX, no time-out mechanism is implemented. Therefore, RX_trig_thresh must be left at 0, or there will be data left in the FIFO that is not fetched by the DMA controller.*
11007038	**FIFO_ADDR_CLR**	16	FIFO address clear register
11007040	**IO_CONFIG**	16	**IO config register** This register is used to configure the I/O for the SDA and SCL lines to select between normal I/O mode, or open-drain mode to support wired-and bus.
11007044	**MULTIMAS**	16	**Multiple I2C masters**

Address	Name	Width	Register function
			This registers contains options for supporting multi-master features.
11007048	**HS**	16	**High speed mode register** This register contains options for supporting high speed operation features. Each HS half pulse width, i.e. each high or low pulse, is equal to (hs_step_cnt_div+1)*(hs_sample_cnt_div+1)*1/4*BASE_CLOCK_PERIOD
11007050	**SOFTRESET**	16	Soft reset register
11007054	**HW_DCM_EN**	16	HW DCM enable
11007064	**DEBUGSTAT**	16	Debug status register
11007068	**DEBUGCTRL**	16	Debug control register

11007000 DATA_PORT Data Port Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DATA_PORT							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0	**DATA_PORT**	DATA_PORT	**FIFO access port** During master write sequences (slave_addr[0] = 0), this port can be written by APB, and during master read sequences (slave_addr[0] = 1), this port can be read by APB. *Note: Slave_addr must be set correctly before accessing FIFO.* For debugging only: If the fifo_apb_debug bit is set, FIFO can be read and written by the APB.

11007004 SLAVE_ADDR Slave Address Register 00BF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SLAVE_ADDR							
Type									RW							
Reset									1	0	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
7:0	**SLAVE_ADDR**	SLAVE_ADDR	**Specifies the slave address of the device to be accessed** Bit 0 is defined by the I2C protocol as a bit that indicates the direction of transfer. 0: Master writer 1: Master read

11007008 INTR_MASK Interrupt Mask Register 00FF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SPARE				MAS	MAS	MAS	MAS

11007008 INTR_MASK Interrupt Mask Register 00FF

														K_ARB_LOST	K_HS_NACKERR	K_ACKERR	K_TRANSAC_COMP
Type									RW					RW	RW	RW	RW
Reset									1	1	1	1		1	1	1	1

Overview: This register provides masks for the corresponding interrupt sources as indicated in intr_stat register. 1 = allow interrupt; 0 = disable interrupt Note that while disabled, the corresponding interrupt will not be asserted. However, intr_stat will still be updated with the status, i.e. mask does not affect intr_stat register values.

Bit(s)	Mnemonic	Name	Description
7:4	**SPARE**	SPARE	Reserved
3	**MASK_ARB_LOST**	MASK_ARB_LOST	Setting this value to 0 will mask ARB_LOST interrupt signal.
2	**MASK_HS_NACKERR**	MASK_HS_NACKERR	Setting this value to 0 will mask HS_NACKERR interrupt signal.
1	**MASK_ACKERR**	MASK_ACKERR	Setting this value to 0 will mask ACK_ERR interrupt signal.
0	**MASK_TRANSAC_COMP**	MASK_TRANSAC_COMP	Setting this value to 0 will mask TRANSAC_COMP interrupt signal.

1100700C INTR_STAT Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ARB_LOST	HS_NACKERR	ACKERR	TRANSAC_COMP
Type													W1C	W1C	W1C	W1C
Reset													0	0	0	0

Overview: When an interrupt is issued by the I2C controller, this register will need to be read by MCU to determine the cause for the interrupt. After this status has been read and appropriate actions are taken, the corresponding interrupt source will need to be written 1 to clear.

Bit(s)	Mnemonic	Name	Description
3	**ARB_LOST**	ARB_LOST	This status is asserted if the I2C controller loses arbitration.
2	**HS_NACKERR**	HS_NACKERR	This status is asserted if HS master code NACK error detection is enabled. If enabled, HS master code NACK err will cause transaction to end and stop will be issued.
1	**ACKERR**	ACKERR	This status is asserted if ACK error detection is enabled. If enabled, ACKERR will cause transaction to end and stop will be issued.
0	**TRANSAC_COMP**	TRANSAC_COMP	This status is asserted when a transaction is completed successfully.

11007010 CONTROL Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										TRANSFER_LEN_CHANGE	ACKERR_DET_EN	DIR_CHANGE	CLK_EXT_EN	DMA_EN	RS_STOP	
Type										RW	RW	RW	RW	RW	RW	
Reset										0	0	0	0	0	0	

Bit(s)	Mnemonic	Name	Description
6	**TRANSFER_LEN_CHANGE**	TRANSFER_LEN_CHANGE	**Specifies whether or not to change the transfer length after the fist transfer is completed** If enabled, the transfers after the first transfer will use the transfer_len_aux parameter.
5	**ACKERR_DET_EN**	ACKERR_DET_EN	**Enables slave ACK error detection** When enabled, if slave ACK error is detected, the master shall terminate the transaction by issuing a STOP condition and then asserts the ACKERR interrupt. MCU handles this case appropriately and then resets the FIFO address before reissuing transaction. If this option is disabled, the controller will ignore slave ACK error and keep on scheduled transaction. 0: Disable 1: Enable
4	**DIR_CHANGE**	DIR_CHANGE	**Combined transfer format, where the direction of transfer is to be changed from write to read after the FIRST RS condition** *Note: When set to 1, the transfers after the direction change will be based on the transfer_len_aux parameter.* 0: Disable 1: Enable
3	**CLK_EXT_EN**	CLK_EXT_EN	**I2C spec allows slaves to hold the SCL line low if it is not yet ready for further processing.** Therefore, if this bit is set to 1, the master controller will enter a high wait state until the slave releases the SCL line.
2	**DMA_EN**	DMA_EN	**By default, this is disabled, and the FIFO data shall be manually prepared by MCU.** This default setting should be used for transfer sizes of smaller than 8 data bytes and no multiple transfer is configured. When enabled, DMA requests are turned on, and the FIFO data should be prepared in memory.
1	**RS_STOP**	RS_STOP	**In LS/FS mode, this bit affects multi-transfer transaction only.** It controls whether or not the REPEATED-START condition is used between transfers. The last ending transfer always ends with a STOP. In HS mode, this bit must be set to 1. 0: Use STOP 1: Use REPEATED-START

11007014 TRANSFER_LEN Transfer Length Register (Number of Bytes per Transfer) 0101

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				TRANSFER_LEN_AUX					TRANSFER_LEN							
Type				RW					RW							

11007014 TRANSFER_LEN Transfer Length Register (Number of Bytes per Transfer) 0101

Reset				0	0	0	0	1	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
12:8	**TRANSFER_LEN_AUX**	TRANSFER_LEN_AUX	**This field is valid only when dir_change is set to 1.** Indicates the number of data bytes to be transferred in 1 transfer unit (excluding slave address byte) for the transfers following the direction change. That is, if dir_change =1, then the first write transfer length will depend on transfer_len, while the second read transfer length depends on transfer_len_aux. Dir change is always after the first transfer. *Note: The value must be set to be bigger than 1; otherwise no transfer will take place.*
7:0	**TRANSFER_LEN**	TRANSFER_LEN	**Indicates the number of data bytes to be transferred in 1 transfer unit (excluding slave address byte)** *Note: The value must be set to be bigger than 1; otherwise no transfer will take place.*

11007018 TRANSAC_LEN Transaction Length Register (Number of Transfers per Transaction) 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									TRANSAC_LEN							
Type									RW							
Reset									0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
7:0	**TRANSAC_LEN**	TRANSAC_LEN	**Indicates the number of transfers to be transferred in 1 transaction** *Note: The value must be set to be bigger than 1; otherwise no transfer will take place.*

1100701C DELAY_LEN Inter Delay Length Register 0002

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DELAY_LEN							
Type									RW							
Reset									0	0	0	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
7:0	**DELAY_LEN**	DELAY_LEN	**Sets up wait delay between consecutive transfers when RS_STOP bit is set to 0** Unit: Half the pulse width)

11007020 TIMING Timing Control Register 1303

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DATA	DATA_READ_TIME				SAMPLE_CNT_DIV					STEP_CNT_DIV					

11007020 TIMING Timing Control Register 1303

	_READ_ADJ															
Type	RW	RW				RW					RW					
Reset	0	0	0	1		0	1	1			0	0	0	0	1	1

Overview: LS/FS only. This register is used to control the output waveform timing. Each half pulse width, i.e. each high or low pulse, is equal to (step_cnt_div+1)*(sample_cnt_div+1)*1/4*BASE_CLOCK_PERIOD.

Bit(s)	Mnemonic	Name	Description
15	**DATA_READ_ADJ**	DATA_READ_ADJ	When set to 1, data latch in sampling time during master reads are adjusted according to the DATA_READ_TIME value. Otherwise, by default, the data are latched in at half of the high pulse width point. This value must be set to be smaller than or equal to half the high pulse width.
14:12	**DATA_READ_TIME**	DATA_READ_TIME	This value is valid only when DATA_READ_ADJ is set to 1. This can be used to adjust so that the data are latched in at earlier sampling points (assuming data are settled by then)
10:8	**SAMPLE_CNT_DIV**	SAMPLE_CNT_DIV	Used for LS/FS only. This adjusts the width of each sample. (sample width = sample_cnt_div*1/4 BASE_CLOCK_PERIOD)
5:0	**STEP_CNT_DIV**	STEP_CNT_DIV	Specifies the number of samples per half pulse width, i.e. each high or low pulse

11007024 START Start Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																START
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**START**	START	**Starts the transaction on the bus** It is auto de-asserted at the end of the transaction.

11007028 EXT_CONF Extension Configuration Register 1800

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXT_TIME															EXT_EN
Type	RW															RW
Reset	0	0	0	1	1	0	0	0								0

Bit(s)	Mnemonic	Name	Description
15:8	**EXT_TIME**	EXT_TIME	**Configurable extension time of start condition** Time unit: 1/4 BASE_CLOCK_PERIOD *Note: The max. value is {0xFF - SAMPLE_CNT_DIV}.*
0	**EXT_EN**	EXT_EN	**Used for standard mode only (baud rate is up to 100kHz)** This option decides to perform the extension of start /stop condition.

Bit(s)	Mnemonic	Name	Description
			If enabled, perform the extension; otherwise not. 0: Disable 1: Enable

11007030 FIFO_STAT FIFO Status Register 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RD_ADDR				WR_ADDR				FIFO_OFFSET						WR_FULL	RD_EMPTY
Type	RU				RU				RU						RU	RU
Reset	0	0	0	0	0	0	0	0	0	0	0	0			0	1

Bit(s)	Mnemonic	Name	Description
15:12	**RD_ADDR**	RD_ADDR	**Current RD address pointer** Only bit [2:0] has physical meaning.
11:8	**WR_ADDR**	WR_ADDR	**Current WR address pointer** Only bit [2:0] has physical meaning.
7:4	**FIFO_OFFSET**	FIFO_OFFSET	**wr_addr[3:0] - rd_addr[3:0]**
1	**WR_FULL**	WR_FULL	**Indicates FIFO is full**
0	**RD_EMPTY**	RD_EMPTY	**Indicates FIFO is empty**

11007034 FIFO_THRESH FIFO Thresh Register 0700

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						TX_TRIG_THRESH								RX_TRIG_THRESH		
Type						RW								RW		
Reset						1	1	1						0	0	0

Overview: For debugging only. By default, these values do not need to be adjusted. Note that for RX, no time-out mechanism is implemented. Therefore, RX_trig_thresh must be left at 0, or there will be data left in FIFO that is not fetched by the DMA controller.

Bit(s)	Mnemonic	Name	Description
10:8	**TX_TRIG_THRESH**	TX_TRIG_THRESH	When Tx FIFO level is below this value, Tx DMA request is asserted.
2:0	**RX_TRIG_THRESH**	RX_TRIG_THRESH	When Rx FIFO level is above this value, Rx DMA request is asserted.

11007038 FIFO_ADDR_CLR FIFO Address Clear Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																FIFO_ADDR_CLR

11007038 **FIFO_ADDR_CLR** **FIFO Address Clear Register** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Type																WO
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**FIFO_ADDR_CLR**	FIFO_ADDR_CLR	When written with 1'b1, a 1 pulse fifo_addr_clr is generated to clear the FIFO address to back to 0.

11007040 **IO_CONFIG** **IO Config Register** **0003**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													**IDLE_OE_EN**	**IO_SYNC_EN**	**SDA_IO_CONFIG**	**SCL_IO_CONFIG**
Type													RW	RW	RW	RW
Reset													0	0	1	1

Overview: This register is used to configure the I/O for the SDA and SCL lines to select between normal I/O mode, or open-drain mode to support wired-and bus.

Bit(s)	Mnemonic	Name	Description
3	**IDLE_OE_EN**	IDLE_OE_EN	0: Does not drive bus in idle state 1: Drive bus in idle state
2	**IO_SYNC_EN**	IO_SYNC_EN	**For debugging only** When set to 1, SCL and SDA inputs will be first dual synced by bclk_ck. This should not be needed. Only reserved for debugging.
1	**SDA_IO_CONFIG**	SDA_IO_CONFIG	0: Normal tristate I/O mode 1: Open-drain mode
0	**SCL_IO_CONFIG**	SCL_IO_CONFIG	0: Normal tristate I/O mode 1: Open-drain mode

11007044 **MULTIMAS** **Multiple I2C Masters** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														**BUS_DET_EN**	**CLK_SYNC_EN**	**ARB_EN**
Type														RW	RW	RW
Reset														0	0	0

Overview: This registers contains options for supporting multi-master features.

Bit(s)	Mnemonic	Name	Description
2	**BUS_DET_EN**	BUS_DET_EN	When enabled, bus status is monitored and if bus is currently busy, no new transaction can proceed.
1	**CLK_SYNC_EN**	CLK_SYNC_EN	When enabled, clk synchronization will be performed.
0	**ARB_EN**	ARB_EN	When enabled, multi-master arbitration will be performed.

11007048 HS High Speed Mode Register 0102

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		HS_SAMPLE_CNT_DIV				HS_STEP_CNT_DIV				MASTER_CODE					HS_NACKERR_DET_EN	HS_EN
Type		RW				RW				RW					RW	RW
Reset		0	0	0		0	0	1		0	0	0			1	0

Overview: This register contains options for supporting high speed operation features Each HS half pulse width, i.e. each high or low pulse, is equal to step_cnt_div*(sample_cnt_div*1/16.25MHz)

Bit(s)	Mnemonic	Name	Description
14:12	**HS_SAMPLE_CNT_DIV**	HS_SAMPLE_CNT_DIV	When the high-speed mode is entered after the master code transfer is completed, the sample width will become dependent on this parameter.
10:8	**HS_STEP_CNT_DIV**	HS_STEP_CNT_DIV	When the high-speed mode is entered after the master code transfer is completed, the number of samples per half pulse width will become dependent on this value.
6:4	**MASTER_CODE**	MASTER_CODE	This is the 3-bit programmable value for the master code to be transmitted.
1	**HS_NACKERR_DET_EN**	HS_NACKERR_DET_EN	**Enables NACKERR detection during the master code transmission** When enabled, if NACK is not received after master code is transmitted, the transaction will be terminated with a STOP condition.
0	**HS_EN**	HS_EN	**Enables high-speed transaction** *Note: rs_stop must be set to 1.*

11007050 SOFTRESET Soft Reset Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SOFT_RESET
Type																WO
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**SOFT_RESET**	SOFT_RESET	When written with 1'b1, a 1 pulse soft reset is used as synchronous reset to reset the I2C internal hardware circuits.

11007054 HW_DCM_EN HW DCM Enable 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DCM_EN
Type																RW
Reset																1

Bit(s)	Mnemonic	Name	Description
0	**DCM_EN**	DCM_EN	**Enables HW DCM function** Default is enable. 0: Disable 1: Enable

11007064 DEBUGSTAT Debug Status Register 0020

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									BUS_BUSY	MASTER_WRITE	MASTER_READ	MASTER_STATE				
Type									RU	RU	RU	RU				
Reset									0	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7	**BUS_BUSY**	BUS_BUSY	**For debugging only** Valid when bus_detect_en is 1. bus_busy = 1 indicates a start transaction has been detected and no stop condition has been detected yet.
6	**MASTER_WRITE**	MASTER_WRITE	**For debugging only** 1: Current transfer is in the master write dir.
5	**MASTER_READ**	MASTER_READ	**For debugging only** 1: Current transfer is in the master read dir.
4:0	**MASTER_STATE**	MASTER_STATE	**For debugging only: Reads back the current master_state.** 0: Idle state 1: I2C master is preparing sending out the start bit, SCL=1, SDA=1. 2: I2C master is sending out the start bit, SCL=1, SDA=0. 3: I2C master/slave is preparing transmitting data bit, SCL=0, SDA=data bit (data bit can be changed when SCL=0). 4: I2C master/slave is transmitting data bit, SCL=1, SDA=data bit (data bit is stable when SCL=1). 5: I2C master/slave is preparing transmitting the ACK bit, SCL=0, SDA=ack (The ACK bit can be changed when SCL=0). 6: I2C master/slave is transmitting the ACK bit, SCL=1, SDA=0 (ack bit is stable when SCL=1). 7: I2C master is preparing sending out stop bit or repeated-start bit, SCL=0, SDA=0/1 (0: Stop bit; 1: Repeated-start bit). 8: I2C master is sending out stop bit or repeated-start bit, SCL=1, SDA=1/0 (1: Stop bit; 0: Repeated-start bit). 9: I2C master is in delay start between two transfers, SCL=1, SDA=1. 10: I2C master is in FIFO wait state; For writing transaction, it means FIFO is empty and I2C master is waiting for DMA controller writing data into FIFO; For reading transaction, it means FIFO is full and I2C master is waiting for DMA controller reading data from FIFO, SCL=0, SDA=don't care. 12: I2C master is preparing sending out data bit of master code. This state is used only in high-speed transaction, SCL=0, SDA=data bit of master code (data bit of master code can be changed when SCL=0). 13: I2C master is sending out data bit of master code. This state is used only in high-speed transaction, SCL=1, SDA=data bit of

Bit(s)	Mnemonic	Name	Description
			master code (data bit of master code is stable when SCL=1). 14: I2C master/slave is preparing transmitting the NACK bit, SCL=0, SDA=nack bit (The NACK bit can be changed when SCL=0); This state is used only in high-speed transaction. 15: I2C master/slave is transmitting the NACK bit, SCL=1, SDA=1; This state is used only in high-speed transaction.

.

11007068 DEBUGCTRL Debug Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															APB_DEBUG_RD	FIFO_APB_DEBUG
Type															WO	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1	**APB_DEBUG_RD**	APB_DEBUG_RD	**Only valid when fifo_apb_debug is set to 1** Writing to this register will generate a 1 pulsed FIFO APB RD signal for reading the FIFO data.
0	**FIFO_APB_DEBUG**	FIFO_APB_DEBUG	**Used for trace 32 debug purposes** When using trace 32, and the memory map is shown, turning this bit on will block the normal APB read access. The APB read access to the FIFO will then be enabled by writing to apb_debug_rd. 0: Disable 1: Enable

10 Pulse-Width Modulator (PWM)

10.1 Introduction

Seven generic pulse-width modulators are implemented to generate pulse sequences with programmable frequency and duration for LCD backlight, charging or other purposes. Before enabling PWM, the pulse sequences must be prepared in the memory or registers. Then PWM will read the pulse sequences to generate random waveform to meet all kinds of applications

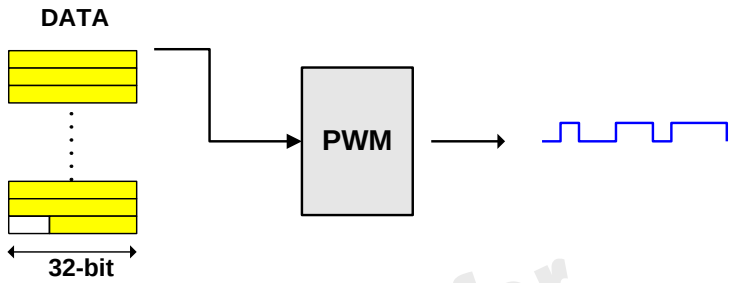


Figure 10-1: Generation procedure of PWM

10.2 Feature list

- Total 5 channel PWM can output to external pins
- Periodical memory and random mode
- Sequential output mode

10.3 Block Diagram

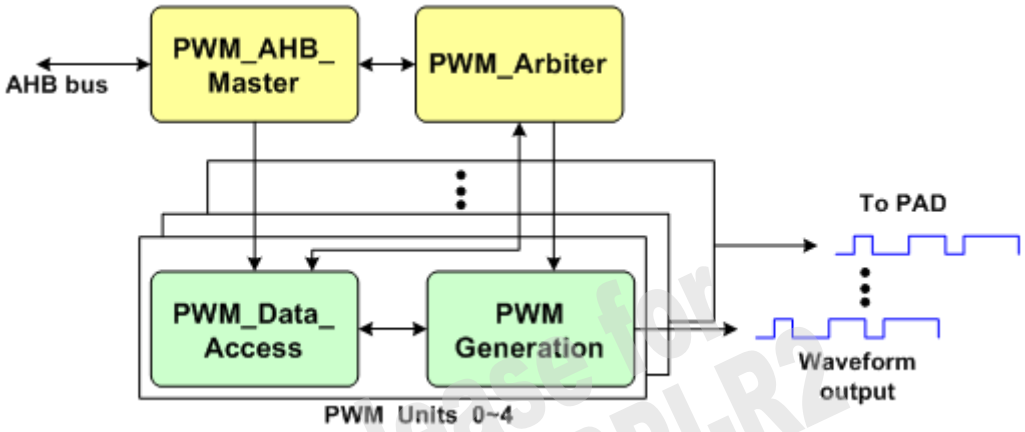


Figure 10-2: Block Diagram of PWM

10.4 Register Definition

Address	Name	Width	Register function
11006000	**PWM_ENABLE**	32	PWM enable register
11006004	**PWM4_DELAY**	32	PWM4 delay duration register
11006008	**PWM5_DELAY**	32	PWM5 delay duration register
1100600C	**PWM6_DELAY**	32	PWM6 delay duration register
11006010	**PWM1_CON**	32	PWM1 control register
11006014	**PWM1_HDURATION**	32	PWM1 high duration register
11006018	**PWM1_LDURATION**	32	PWM1 low duration register
1100601C	**PWM1_GDRUATION**	32	PWM1 guard duration register
11006020	**PWM1_BUF0_BASE_ ADDR**	32	PWM1 buffer 0 base address register
11006024	**PWM1_BUF0_SIZE**	32	PWM1 buffer 0 size register
11006028	**PWM1_BUF1_BASE_ ADDR**	32	PWM1 buffer 1 base address register
1100602C	**PWM1_BUF1_SIZE**	32	PWM1 buffer1 size register
11006030	**PWM1_SEND_DATA0**	32	PWM1 send data 0 register
11006034	**PWM1_SEND_DATA1**	32	PWM1 send data1 register
11006038	**PWM1_WAVE_NUM**	32	PWM1 wave number register
1100603C	**PWM1_DATA_WIDTH**	32	PWM1 data width register
11006040	**PWM1_THRESH**	32	PWM1 thresh register
11006044	**PWM1_SEND_WAVEN UM**	32	PWM1 send wave number register
11006048	**PWM1_VALID**	32	PWM1 valid register
11006050	**PWM2_CON**	32	PWM2 control register
11006054	**PWM2_HDURATION**	32	PWM2 high duration register
11006058	**PWM2_LDURATION**	32	PWM2 low duration register
1100605C	**PWM2_GDRUATION**	32	PWM2 guard duration register
11006060	**PWM2_BUF0_BASE_ ADDR**	32	PWM2 buffer 0 base address register
11006064	**PWM2_BUF0_SIZE**	32	PWM2 buffer 0 size register
11006068	**PWM2_BUF1_BASE_ ADDR**	32	PWM2 buffer 1 base address register
1100606C	**PWM2_BUF1_SIZE**	32	PWM2 buffer 1 size register
11006070	**PWM2_SEND_DATA0**	32	PWM2 send data 0 register
11006074	**PWM2_SEND_DATA1**	32	PWM2 send data 1 register
11006078	**PWM2_WAVE_NUM**	32	PWM2 wave number register
1100607C	**PWM2_DATA_WIDTH**	32	PWM2 data width register
11006080	**PWM2_THRESH**	32	PWM2 thresh register

Address	Name	Width	Register function
11006084	**PWM2_SEND_WAVENUM**	32	PWM2 send wave number register
11006088	**PWM2_VALID**	32	PWM2 valid register
11006090	**PWM3_CON**	32	PWM3 control register
11006094	**PWM3_HDURATION**	32	PWM3 high duration register
11006098	**PWM3_LDURATION**	32	PWM3 low duration register
1100609C	**PWM3_GDRUATION**	32	PWM3 guard duration register
110060A0	**PWM3_BUF0_BASE_ADDR**	32	PWM3 buffer 0 base address register
110060A4	**PWM3_BUF0_SIZE**	32	PWM3 buffer 0 size register
110060A8	**PWM3_BUF1_BASE_ADDR**	32	PWM3 buffer 1 base address register
110060AC	**PWM3_BUF1_SIZE**	32	PWM3 buffer 1 size register
110060B0	**PWM3_SEND_DATA0**	32	PWM3 send data 0 register
110060B4	**PWM3_SEND_DATA1**	32	PWM3 send data 1 register
110060B8	**PWM3_WAVE_NUM**	32	PWM3 wave number register
110060BC	**PWM3_DATA_WIDTH**	32	PWM3 data width register
110060C0	**PWM3_THRESH**	32	PWM3 thresh register
110060C4	**PWM3_SEND_WAVENUM**	32	PWM3 send wave number register
110060C8	**PWM3_VALID**	32	PWM3 valid register
110060D0	**PWM4_CON**	32	PWM4 control register
110060D4	**PWM4_HDURATION**	32	PWM4 high duration register
110060D8	**PWM4_LDURATION**	32	PWM4 low duration register
110060DC	**PWM4_GDRUATION**	32	PWM4 guard duration register
110060E0	**PWM4_BUF0_BASE_ADDR**	32	PWM4 buffer 0 base address register
110060E4	**PWM4_BUF0_SIZE**	32	PWM4 buffer 0 size register
110060E8	**PWM4_BUF1_BASE_ADDR**	32	PWM4 buffer 1 base sddress register
110060EC	**PWM4_BUF1_SIZE**	32	PWM4 buffer1 size register
110060F0	**PWM4_SEND_DATA0**	32	PWM4 send data 0 register
110060F4	**PWM4_SEND_DATA1**	32	PWM4 send data 1 register
110060F8	**PWM4_WAVE_NUM**	32	PWM4 wave number register
110060FC	**PWM4_SEND_WAVENUM**	32	PWM4 send wave number register
11006100	**PWM4_DATA_WIDTH**	32	PWM4 data width register
11006104	**PWM4_THRESH**	32	PWM4 thresh register
11006108	**PWM4_VALID**	32	PWM4 valid register
11006110	**PWM5_CON**	32	PWM5 control register
11006114	**PWM5_HDURATION**	32	PWM5 high duration register

Address	Name	Width	Register function
11006118	**PWM5_LDURATION**	32	PWM5 low duration register
1100611C	**PWM5_GDRUATION**	32	PWM5 guard duration register
11006120	**PWM5_BUF0_BASE_ADDR**	32	PWM5 buffer 0 base address register
11006124	**PWM5_BUF0_SIZE**	32	PWM5 buffer 0 size register
11006128	**PWM5_BUF1_BASE_ADDR**	32	PWM5 buffer 1 base address register
1100612C	**PWM5_BUF1_SIZE**	32	PWM5 buffer 1 size register
11006130	**PWM5_SEND_DATA0**	32	PWM5 send data 0 register
11006134	**PWM5_SEND_DATA1**	32	PWM5 send data 1 register
11006138	**PWM5_WAVE_NUM**	32	PWM5 wave number register
1100613C	**PWM5_SEND_WAVENUM**	32	PWM5 send wave number register
11006140	**PWM5_DATA_WIDTH**	32	PWM5 data width register
11006144	**PWM5_THRESH**	32	PWM5 thresh register
11006148	**PWM5_VALID**	32	PWM5 valid register
110061CC	**PWM_LOOP_BACK_TEST**	32	PWM loop back test
110061D0	**PWM_3DLCM**	32	PWM support for 3D LCM base pwm2 , select pwm3 , pwm4 ,pwm5 same as pwm2 or inversion of pwm2
11006200	**PWM_INT_ENABLE**	32	PWM interrupt enable register
11006204	**PWM_INT_STATUS**	32	PWM interrupt status register
11006208	**PWM_INT_ACK**	32	PWM interrupt acknowledge register
1100620C	**PWM_EN_STATUS**	32	PWM enable status register

11006000 **PWM_ENABLE** **PWM Enable Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															**PWM_TEST_SEL**	**PWM_SEQ_MODE**
Type															RW	RW
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												**PWM5_EN**	**PWM4_EN**	**PWM3_EN**	**PWM2_EN**	**PWM1_EN**
Type												RW	RW	RW	RW	RW
Reset												0	0	0	0	0

Bit(s)	Name	Description
17	PWM_TEST_SEL	**Set to 1 to enable the switch of the PWM output signal between PWM unit 1, PWM unit 5**

Bit(s)	Name	Description
		The default (0) setting is to select the output of PWM unit 5. If set to 1, the output of PWM unit1 will be selected instead of PWM unit 5.
16	PWM_SEQ_MODE	**Set to 1 to enable PWM3, PWM4, PWM5 sequential delay mode** In this mode, PWM3 starts first, and after PWM4_DELAY_TIME, PWM4 will start. After PWM4 starts, PWM5 will start after PWM5_DELAY_TIME. *Note: The output of PWM_SEQ_MODE is started after PWM3 is enabled. PWM_SEQ_MODE should be set before PWM4, PWM5 are enabled or at the same time. This mode does not work when PWM3 is set at OLD_PWM_MODE and CLKSEL = 1.*
4	PWM5_EN	**Set to 1 to enable PWM5**
3	PWM4_EN	**Set to 1 to enable PWM4**
2	PWM3_EN	**Set to 1 to enable PWM3**
1	PWM2_EN	**Set to 1 to enable PWM2**
0	PWM1_EN	**Set to 1 to enable PWM1**

11006004 PWM4_DELAY PWM4 Delay Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																DELAY_CLKSEL
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PWM4_DELAY_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16	DELAY_CLKSEL	**Clock unit of PWM4_DELAY_DURATION** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
15:0	PWM4_DELAY_DURATION	**Time difference between PWM3 and PWM4**

11006008 PWM5_DELAY PWM5 Delay Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																DELAY_CLKSEL
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PWM5_DELAY_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16	DELAY_CLKSEL	**Clock unit of PWM5_DELAY_DURATION**

Bit(s)	Name	Description
		0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
15:0	PWM5_DELAY_DURATION	**Time difference between PWM4 and PWM5**

1100600C **PWM6_DELAY** **PWM6 Delay Duration Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																DELAY_CLKSEL
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PWM6_DELAY_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16	DELAY_CLKSEL	**Clock unit of PWM6_DELAY_DURATION** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
15:0	PWM6_DELAY_DURATION	**Time difference between PWM5 and PWM6**

11006010 **PWM1_CON** **PWM1 Control Register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	MODE	SRCSEL		CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RW	RW		RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	OLD_PWM_MODE	**Uses old PWM mode** *Note: Using old PWM mode also means using periodical mode. Therefore, SRCSEL and MODE are ignored in this situation. Only old PWM mode with 32kHz clock source(however ,cannot work in sleep mode).* 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Stop bit position for source data in periodical mode** In FIFO mode, it is used to indicate the stop bit position in the total 64 bits(1 ~63;62 is not supported). In the period memory mode, it is for the stop bit position in the last 32 bits (1 ~31;30 is not supported).

Bit(s)	Name	Description
8	GUARD_VALUE	**PWM1 output value when in guard time**
7	IDLE_VALUE	**PWM1 output value when in idle state**
6	MODE	**Selects random generator mode** 0: Periodical PWM mode 1: Random PWM mode
5	SRCSEL	**Selects PWM1 data source** 0: FIFO mode 1: Memory mode
3	CLKSEL	**Selects PWM1 clock** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
2:0	CLKDIV	**Selects PWM1 clock scale** 000b: CLK Hz 001b: CLK/2Hz 010b: CLK/4Hz 011b: CLK/8Hz 100b: CLK/16Hz 101b: CLK/32Hz 110b: CLK/64Hz 111b: CLK/128Hz

11006014 PWM1_HDURATION PWM1 High Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	HDURATION	**PWM1 pulse duration based on the current clock when PWM output is high** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

11006018 PWM1_LDURATION PWM1 Low Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	LDURATION	**PWM1 pulse duration based on the current clock when PWM output is low** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

1100601C PWM1_GDRUATION PWM1 Guard Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GUARD_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	GUARD_DURATION	**Guarding interval between individual waveforms in fifo mode and period memory mode; output is decided by GUARD_VALUE** If it equals N, program N-1 in this register. *Note: If this duration is 0, it means there is no guarding interval. The guard duration of old mode should be set 0.*

11006020 PWM1_BUF0_BASE_ADDR PWM1 Buffer 0 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF0_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF0_BS_ADDR	Base address of memory buffer 0 for PWM1's waveform data

11006024 PWM1_BUF0_SIZE PWM1 Buffer 0 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
15:0	BUF0_SIZE	**Length of the waveform data in memory buffer 0 that PWM1 should generate** If it equals N, program N-1 in this register. *Note: The size is in unit of 32-bit data.*

11006028 PWM1_BUF1_BASE_ADDR PWM1 Buffer 1 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF1_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF1_BS_ADDR	**Base address of memory buffer 1 for PWM1's waveform data** *Note: Memory buffer 1 is useless in periodical mode.*

1100602C PWM1_BUF1_SIZE PWM1 Buffer 1 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF1_SIZE	**Length of the waveform data in memory buffer 1 that PWM1 should generate** If it equals N, program N-1 in this register.

11006030 PWM1_SEND_DATA0 PWM1 Send Data 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM1 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

11006034 PWM1_SEND_DATA1 PWM1 Send Data 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM1 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

11006038 PWM1_WAVE_NUM PWM1 Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAVE_NUM															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM1 will generate from the pulse data repeatedly** *Note: If WAVE_NUM = 0, the waveform generation will not stop until it is disabled.*

1100603C PWM1_DATA_WIDTH PWM1 Data Width Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DATA_WIDTH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	DATA_WIDTH	PWM1 pulse data width in old PWM mode

11006040 PWM1_THRESH PWM1 Thresh Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				THRESH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	THRESH	PWM1 pulse data high/low switching threshold in old PWM mode

11006044 PWM1_SEND_WAVENUM PWM1 Send Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_WAVENUM															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	SEND_WAVENUM	The number by which PWM1 has already generated from the specified data source in the periodical mode

11006048 PWM1_VALID PWM1 Valid Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													BUF1_VALID_WEN	BUF1_VALID	BUF0_VALID_WEN	BUF0_VALID
Type													WO	RW	WO	RW
Reset													0	0	0	0

Bit(s)	Name	Description
3	BUF1_VALID_WEN	This bit must be set to modify BUF1_VALID.
2	BUF1_VALID	The valid status is used to indicate pulse data when memory buffer 1 is ready.
1	BUF0_VALID_WEN	This bit must be set to modify BUF0_VALID.
0	BUF0_VALID	The valid status is used to indicate pulse data when memory buffer 0 is

Bit(s)	Name	Description
		ready.

11006050 PWM2_CON PWM2 Control Register 00007E00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	MODE	SRCSEL		CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RW	RW		RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	OLD_PWM_MODE	**Uses old PWM mode** *Note: Using old PWM mode also means using periodical mode. Therefore, SRCSEL and MODE are ignored in this situation. Only old PWM mode with 32kHz clock source (however ,cannot work in sleep mode).* 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Stop bit position for source data in periodical mode** In FIFO mode, it is used to indicate the stop bit position in the total 64 bits (1 ~63; 62 is not supported). In the period memory mode, it is for the stop bit position in the last 32 bits (1 ~ 31; 30 is not supported).
8	GUARD_VALUE	**PWM2 output value when in guard time**
7	IDLE_VALUE	**PWM2 output value when in idle state**
6	MODE	**Selects random generator mode** 0: Periodical PWM mode 1: Random PWM mode
5	SRCSEL	**Selects PWM2 data source** 0: FIFO mode 1: Memory mode
3	CLKSEL	**Selects PWM1 clock** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
2:0	CLKDIV	**Selects PWM2 clock scale** 000b: CLK Hz 001b: CLK/2Hz 010b: CLK/4Hz 011b: CLK/8Hz 100b: CLK/16Hz 101b: CLK/32Hz 110b: CLK/64Hz 111b: CLK/128Hz

11006054 PWM2_HDURATION PWM2 High Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	HDURATION	**PWM2 pulse duration based on the current clock when PWM output is high** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

11006058 PWM2_LDURATION PWM2 Low Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	LDURATION	**PWM2 pulse duration based on the current clock when PWM output is low** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

1100605C PWM2_GDRUATION PWM2 Guard Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GUARD_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	GUARD_DURATION	**Guarding interval between individual waveforms in fifo mode and period memory mode; output is decided by GUARD_VALUE** If it equals to N, program N-1 in this register. *Note: If this duration is 0, it means there is no guarding interval. The guard duration of old mode should be set 0*

11006060 **PWM2_BUF0_BASE_ADDR** **PWM2 Buffer 0 Base Address Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF0_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF0_BS_ADDR	Base address of memory buffer 0 for PWM2's waveform data

11006064 **PWM2_BUF0_SIZE** **PWM2 Buffer 0 Size Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF0_SIZE	**Length of the waveform data in memory buffer 0 that PWM2 should generate** If it equals to N, program N-1 in this register.

11006068 **PWM2_BUF1_BASE_ADDR** **PWM2 Buffer 1 Base Address Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF1_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF1_BS_ADDR	**Base address of memory buffer 1 for PWM2's waveform data.** *Note: Memory buffer 1 is useless in periodical mode.*

1100606C **PWM2_BUF1_SIZE** **PWM2 Buffer 1 Size Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

1100606C PWM2_BUF1_SIZE PWM2 Buffer 1 Size Register 00000000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF1_SIZE	**Length of the waveform data in memory buffer 1 that PWM2 should generate** If it equals to N, program N-1 in this register.

11006070 PWM2_SEND_DATA0 PWM2 Send Data 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM2 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

11006074 PWM2_SEND_DATA1 PWM2 Send Data1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM2 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

11006078 PWM2_WAVE_NUM PWM2 Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																

11006078 PWM2_WAVE_NUM PWM2 Wave Number Register 00000000

Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAVE_NUM															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM2 will generate from the pulse data repeatedly** *Note: If WAVE_NUM = 0, the waveform generation will not stop until it is disabled.*

1100607C PWM2_DATA_WIDTH PWM2 Data Width Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DATA_WIDTH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	DATA_WIDTH	PWM2 pulse data width in old PWM mode

11006080 PWM2_THRESH PWM2 Thresh Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				THRESH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	THRESH	PWM2 pulse data high/low switching threshold in old PWM mode

11006084 PWM2_SEND_WAVENUM PWM2 Send Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_WAVENUM															
Type	RO															

11006084 **PWM2_SEND_WAVENUM** **PWM2 Send Wave Number Register** **00000000**

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	SEND_WAVENUM	The number by which PWM2 has already generated from the specified data source in the periodical mode

11006088 **PWM2_VALID** **PWM2 Valid Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													BUF1_VALID_WEN	BUF1_VALID	BUF0_VALID_WEN	BUF0_VALID
Type													WO	RW	WO	RW
Reset													0	0	0	0

Bit(s)	Name	Description
3	BUF1_VALID_WEN	This bit must be set to modify BUF1_VALID.
2	BUF1_VALID	The valid status is used to indicate pulse data when memory buffer 1 is ready.
1	BUF0_VALID_WEN	This bit must be set to modify BUF0_VALID.
0	BUF0_VALID	The valid status is used to indicate pulse data when memory buffer0 is ready.

11006090 **PWM3_CON** **PWM3 Control Register** **00007E00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	MODE	SRCSEL		CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RW	RW		RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	OLD_PWM_MODE	**Uses old PWM mode** *Note: Using old PWM mode also means using periodical mode. Therefore, SRCSEL and MODE are ignored in this situation. Only old PWM mode with 32kHz clock source (however ,cannot work in sleep mode).* 0: New PWM mode 1: Old PWM mode

Bit(s)	Name	Description
14:9	STOP_BITPOS	**Stop bit position for source data in periodical mode** In FIFO mode, it is used to indicate the stop bit position in the total 64 bits(1 ~63;62 is not supported). In the period memory mode, it is for the stop bit position in the last 32 bits (1 ~31;30 is not supported).
8	GUARD_VALUE	**PWM3 output value when in guard time**
7	IDLE_VALUE	**PWM3 output value when in idle state**
6	MODE	**Selects random generator mode** 0: Periodical PWM mode 1: Random PWM mode
5	SRCSEL	**Selects PWM3 data source** 0: FIFO mode 1: Memory mode
3	CLKSEL	**Selects PWM1 clock** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
2:0	CLKDIV	**Selects PWM3 clock scale** 000b: CLK Hz 001b: CLK/2Hz 010b: CLK/4Hz 011b: CLK/8Hz 100b: CLK/16Hz 101b: CLK/32Hz 110b: CLK/64Hz 111b: CLK/128Hz

11006094 PWM3_HDURATION PWM3 High Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	HDURATION	**PWM3 pulse duration based on the current clock when PWM output is high** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

11006098 PWM3_LDURATION PWM3 Low Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																

11006098 PWM3_LDURATION PWM3 Low Duration Register 00000001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	LDURATION	**PWM3 pulse duration based on the current clock when PWM output is low** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

1100609C PWM3_GDRUATION PWM3 Guard Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GUARD_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	GUARD_DURATION	**Guarding interval between individual waveforms in fifo mode and period memory mode; output is decided by GUARD_VALUE** If it equals N, program N-1 in this register. *Note: If this duration is 0, it means there is no guarding interval. The guard duration of old mode should be set 0*

110060A0 PWM3_BUF0_BASE_ADDR PWM3 Buffer 0 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF0_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF0_BS_ADDR	Base address of memory buffer 0 for PWM3's waveform data

110060A4 PWM3_BUF0_SIZE PWM3 Buffer 0 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																

110060A4 PWM3_BUF0_SIZE PWM3 Buffer 0 Size Register 00000000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF0_SIZE	**Length of the waveform data in memory buffer 0 that PWM3 should generate** If it equals N, program N-1 in this register.

110060A8 PWM3_BUF1_BASE_ADDR PWM3 Buffer 1 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF1_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF1_BS_ADDR	**Base address of memory buffer 1 for PWM3's waveform data** *Note: Memory buffer 1 is useless in periodical mode.*

110060AC PWM3_BUF1_SIZE PWM3 Buffer 1 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF1_SIZE	**Length of the waveform data in memory buffer 1 that PWM3 should generate** If it equals N, program N-1 in this register.

110060B0 PWM3_SEND_DATA0 PWM3 Send Data 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

110060B0 PWM3_SEND_DATA0 PWM3 Send Data 0 Register 00000000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM3 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

110060B4 PWM3_SEND_DATA1 PWM3 Send Data 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM3 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

110060B8 PWM3_WAVE_NUM PWM3 Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAVE_NUM															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM3 will generate from the pulse data repeatedly** *Note: If WAVE_NUM = 0, the waveform generation will not stop until it is disabled.*

110060BC PWM3_DATA_WIDTH PWM3 Data Width Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

110060BC PWM3_DATA_WIDTH PWM3 Data Width Register 00000000

Name				DATA_WIDTH													
Type				RW													
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	DATA_WIDTH	PWM3 pulse data width in old PWM mode

110060C0 PWM3_THRESH PWM3 Thresh Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				THRESH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	THRESH	PWM3 pulse data high/low switching threshold in old PWM mode

110060C4 PWM3_SEND_WAVENUM PWM3 Send Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_WAVENUM															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	SEND_WAVENUM	The number by which PWM3 has already generated from the specified data source in the periodical mode

110060C8 PWM3_VALID PWM3 Valid Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													BUF1_VALID_WEN	BUF1_VALID	BUF0_VALID_WEN	BUF0_VALID
Type													WO	RW	WO	RW
Reset													0	0	0	0

Bit(s)	Name	Description
3	BUF1_VALID_WEN	This bit must be set to modify BUF1_VALID.
2	BUF1_VALID	The valid status is used to indicate pulse data when memory buffer1 is ready.
1	BUF0_VALID_WEN	This bit must be set to modify BUF0_VALID.
0	BUF0_VALID	The valid status is used to indicate pulse data when memory buffer0 is ready.

110060D0 PWM4_CON PWM4 Control Register 00007E00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	MODE	SRCSEL		CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RW	RW		RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	OLD_PM_MODE	**Uses old PWM mode** *Note: Using old PWM mode also means using periodical mode. Therefore, SRCSEL and MODE are ignored in this situation. Only old PWM mode with 32kHz clock source(however ,cannot work in sleep mode).* 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Stop bit position for source data in periodical mode** In FIFO mode, it is used to indicate the stop bit position in the total 64 bits(1 ~63;62 is not supported). In the period memory mode, it is for the stop bit position in the last 32 bits (1 ~31;30 is not supported).
8	GUARD_VALUE	**PWM4 output value when in guard time**
7	IDLE_VALUE	**PWM4 output value when in idle state**
6	MODE	**Selects random generator mode** 0: Periodical PWM mode 1: Random PWM mode
5	SRCSEL	**Selects PWM4 data source** 0: FIFO mode 1: Memory mode
3	CLKSEL	**Selects PWM1 clock** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
2:0	CLKDIV	**Selects PWM4 clock scale** 000b: CLK Hz 001b: CLK/2Hz 010b: CLK/4Hz

Bit(s)	Name	Description
		011b: CLK/8Hz 100b: CLK/16Hz 101b: CLK/32Hz 110b: CLK/64Hz 111b: CLK/128Hz

110060D4 PWM4_HDURATION PWM4 High Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	HDURATION	**PWM4 pulse duration based on the current clock when PWM output is high** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

110060D8 PWM4_LDURATION PWM4 Low Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	LDURATION	**PWM4 pulse duration based on the current clock when PWM output is low** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

110060DC PWM4_GDRUATION PWM4 Guard Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GUARD_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	GUARD_DURATION	**Guarding interval between individual waveforms in fifo mode and period memory mode; output is decided by GUARD_VALUE** If it equals N, program N-1 in this register. *Note: If this duration is 0, it means there is no guarding interval. The guard duration of old mode should be set 0.*

110060E0 PWM4_BUF0_BASE_ADDR PWM4 Buffer 0 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF0_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF0_BS_ADDR	Base address of memory buffer 0 for PWM4's waveform data

110060E4 PWM4_BUF0_SIZE PWM4 Buffer 0 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF0_SIZE	**Length of the waveform data in memory buffer 0 that PWM4 should generate** If it equals N, program N-1 in this register.

110060E8 PWM4_BUF1_BASE_ADDR PWM4 Buffer 1 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF1_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF1_BS_ADDR	**Base address of memory buffer 1 for PWM4's waveform data** *Note: Memory buffer 1 is useless in periodical mode.*

110060EC PWM4_BUF1_SIZE PWM4 Buffer 1 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF1_SIZE	**Length of the waveform data in memory buffer 1 that PWM4 should generate** If it equals N, program N-1 in this register.

110060F0 PWM4_SEND_DATA0 PWM4 Send Data 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM4 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

110060F4 PWM4_SEND_DATA1 PWM4 Send Data 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM4 local buffe r0 of pulse sequence data to be generated.** *Note: This value should be written only in periodical FIFO mode. In other*

Bit(s)	Name	Description
		modes, this buffer is for internal memory access.

110060F8 PWM4_WAVE_NUM PWM4 Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAVE_NUM															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM4 will generate from the pulse data repeatedly** *Note: If WAVE_NUM = 0, the waveform generation will not stop until it is disabled.*

110060FC PWM4_SEND_WAVENUM PWM4 Send Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_WAVENUM															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	SEND_WAVENUM	The number by which PWM4 has already generated from the specified data source in the periodical mode

11006100 PWM4_DATA_WIDTH PWM4 Data Width Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DATA_WIDTH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	DATA_WIDTH	PWM4 pulse data width in old PWM mode

11006104 PWM4_THRESH PWM4 Thresh Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				THRESH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	THRESH	PWM4 pulse data high/low switching threshold in old PWM mode

11006108 PWM4_VALID PWM4 Valid Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													BUF1_VALID_WEN	BUF1_VALID	BUF0_VALID_WEN	BUF0_VALID
Type													WO	RW	WO	RW
Reset													0	0	0	0

Bit(s)	Name	Description
3	BUF1_VALID_WEN	This bit must be set to modify BUF1_VALID.
2	BUF1_VALID	The valid status is used to indicate pulse data when memory buffer1 is ready.
1	BUF0_VALID_WEN	This bit must be set to modify BUF0_VALID.
0	BUF0_VALID	The valid status is used to indicate pulse data when memory buffer0 is ready.

11006110 PWM5_CON PWM5 Control Register 00007E00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	OLD_PWM_MODE	STOP_BITPOS						GUARD_VALUE	IDLE_VALUE	MODE	SRCSEL		CLKSEL	CLKDIV		
Type	RW	RW						RW	RW	RW	RW		RW	RW		
Reset	0	1	1	1	1	1	1	0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	OLD_PM_MODE	**Uses old PWM mode** *Note: Using old PWM mode also means using periodical mode. Therefore, SRCSEL and MODE are ignored in this situation. Only old PWM mode with*

Bit(s)	Name	Description
		32kHz clock source(however ,cannot work in sleep mode). 0: New PWM mode 1: Old PWM mode
14:9	STOP_BITPOS	**Stop bit position for source data in periodical mode** In FIFO mode, it is used to indicate the stop bit position in the total 64 bits(1 ~63;62 is not supported). In the period memory mode, it is for the stop bit position in the last 32 bits (1 ~31;30 is not supported).
8	GUARD_VALUE	**PWM5 output value when in guard time**
7	IDLE_VALUE	**PWM5 output value when in idle state**
6	MODE	**Selects random generator mode** 0: Periodical PWM mode 1: Random PWM mode
5	SRCSEL	**Selects PWM5 data source** 0: FIFO mode 1: Memory mode
3	CLKSEL	**Selects PWM1 clock** 0: CLK = CLKSRC 1: CLK = CLKSRC/1,625
2:0	CLKDIV	**Selects PWM5 clock scale** 000b: CLK Hz 001b: CLK/2Hz 010b: CLK/4Hz 011b: CLK/8Hz 100b: CLK/16Hz 101b: CLK/32Hz 110b: CLK/64Hz 111b: CLK/128Hz

11006114 PWM5_HDURATION PWM5 High Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	HDURATION	**PWM5 pulse duration based on the current clock when PWM output is high** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

11006118 PWM5_LDURATION PWM5 Low Duration Register 00000001

11006118 PWM5_LDURATION PWM5 Low Duration Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LDURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
15:0	LDURATION	**PWM5 pulse duration based on the current clock when PWM output is low** If duration = N, program N-1 in this register. *Note: The duration of PWM must not be 0.*

1100611C PWM5_GDRUATION PWM5 Guard Duration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GUARD_DURATION															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	GUARD_DURATION	**Guarding interval between individual waveforms in fifo mode and period memory mode; output is decided by GUARD_VALUE** If it equals N, program N-1 in this register. *Note: If this duration is 0, it means there is no guarding interval. The guard duration of old mode should be set 0.*

11006120 PWM5_BUF0_BASE_ADDR PWM5 Buffer 0 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF0_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF0_BS_ADDR	Base address of memory buffer 0 for PWM5's waveform data

11006124 PWM5_BUF0_SIZE PWM5 Buffer 0 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF0_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF0_SIZE	**Length of the waveform data in memory buffer 0 that PWM5 should generate** If it equals N, program N-1 in this register.

11006128 PWM5_BUF1_BASE_ADDR PWM5 Buffer 1 Base Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUF1_BS_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_BS_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	BUF1_BS_ADDR	**Base address of memory buffer 1 for PWM5's waveform data** *Note: Memory buffer 1 is useless in periodical mode.*

1100612C PWM5_BUF1_SIZE PWM5 Buffer 1 Size Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUF1_SIZE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	BUF1_SIZE	**Length of the waveform data in memory buffer 1 that PWM5 should generate** If it equals N, program N-1 in this register.

11006130 PWM5_SEND_DATA0 PWM5 Send Data 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

11006130 PWM5_SEND_DATA0 PWM5 Send Data 0 Register 00000000

Name	SEND_DATA0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA0	**PWM5 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other modes, this buffer is for internal memory access.*

11006134 PWM5_SEND_DATA1 PWM5 Send Data 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SEND_DATA1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_DATA1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	SEND_DATA1	**PWM5 local buffer 0 of pulse sequence data to be generated** *Note: This value should be written only in periodical FIFO mode. In other moded, this buffer is for internal memory access.*

11006138 PWM5_WAVE_NUM PWM5 Wave Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAVE_NUM															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	WAVE_NUM	**The number by which PWM5 will generate from the pulse data repeatedly** *Note: If WAVE_NUM = 0, the waveform generation will not stop until it is disabled.*

1100613C PWM5_SEND_WAVENUM PWM5 Send Wave Number Register 00000000

1100613C **PWM5_SEND_WAVENUM** **PWM5 Send Wave Number Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEND_WAVENUM															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	SEND_WAVENUM	The number by which PWM5 has already generated from the specified data source in the periodical mode

11006140 **PWM5_DATA_WIDTH** **PWM5 Data Width Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DATA_WIDTH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	DATA_WIDTH	PWM5 pulse data width in old PWM mode

11006144 **PWM5_THRESH** **PWM5 Thresh Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				THRESH												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:0	THRESH	PWM5 pulse data high/low switching threshold in old PWM mode

11006148 **PWM5_VALID** **PWM5 Valid Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													BUF1	BUF1	BUF0	BUF0

11006148 PWM5_VALID PWM5 Valid Register 00000000

													_VALID_WEN	_VALID	_VALID_WEN	_VALID
Type													WO	RW	WO	RW
Reset													0	0	0	0

Bit(s)	Name	Description
3	BUF1_VALID_WEN	This bit must be set to modify BUF1_VALID.
2	BUF1_VALID	The valid status is used to indicate pulse data when memory buffer 1 is ready.
1	BUF0_VALID_WEN	This bit must be set to modify BUF0_VALID.
0	BUF0_VALID	The valid status is used to indicate pulse data when memory buffer 0 is ready.

110061CC PWM_LOOP_BACK_TEST PWM Loop Back Test 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												PWM5_STATUS	PWM4_STATUS	PWM3_STATUS	PWM2_STATUS	PWM1_STATUS
Type												RU	RU	RU	RU	RU
Reset												0	0	0	0	0

Bit(s)	Name	Description
4	PWM5_STATUS	PWM5 current status (for loop back test)
3	PWM4_STATUS	PWM4 current status (for loop back test)
2	PWM3_STATUS	PWM3 current status (for loop back test)
1	PWM2_STATUS	PWM2 current status (for loop back test)
0	PWM1_STATUS	PWM1 current status (for loop back test)

110061D0 PWM_3DLCM PWM Support for 3D LCM 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												PWM_3DLCM_2_INV	PWM_3DLCM_1_INV	PWM_3DLCM_0_INV		PWM_3DLCM_EN
Type												RW	RW	RW		RW
Reset												0	0	0		0

Bit(s)	Name	Description

Bit(s)	Name	Description
4	PWM_3DLCM_2_INV	**Set to 1 to make PWM5 inversion of PWM2** 0: PWM5 is the same as PWM2. 1: PWM5 is inversion of PWM2.
3	PWM_3DLCM_1_INV	**Set to 1 to make PWM4 inversion of PWM2** 0: PWM4 is the same as PWM2. 1: PWM4 is inversion of PWM2.
2	PWM_3DLCM_0_INV	**Set to 1 to make PWM3 inversion of PWM2** 0: PWM3 is the same as PWM2. 1: PWM3 is inversion of PWM2.
0	PWM_3DLCM_EN	**Set to 1 to enable PWM for 3D LCM**

11006200 **PWM_INT_ENABLE** **PWM Interrupt Enable Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							**PWM5_INT_UNDERFLOW_EN**	**PWM5_INT_FINISH_EN**	**PWM4_INT_UNDERFLOW_EN**	**PWM4_INT_FINISH_EN**	**PWM3_INT_UNDERFLOW_EN**	**PWM3_INT_FINISH_EN**	**PWM2_INT_UNDERFLOW_EN**	**PWM2_INT_FINISH_EN**	**PWM1_INT_UNDERFLOW_EN**	**PWM1_INT_FINISH_EN**
Type							RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9	PWM5_INT_UNDERFLOW_EN	Set to 1 to enable PWM5 underflow interrupt
8	PWM5_INT_FINISH_EN	Set to 1 to enable PWM5 finish interrupt
7	PWM4_INT_UNDERFLOW_EN	Set to 1 to enable PWM4 underflow interrupt
6	PWM4_INT_FINISH_EN	Set to 1 to enable PWM4 finish interrupt
5	PWM3_INT_UNDERFLOW_EN	Set to 1 to enable PWM3 underflow interrupt
4	PWM3_INT_FINISH_EN	Set to 1 to enable PWM3 finish interrupt
3	PWM2_INT_UNDERFLOW_EN	Set to 1 to enable PWM2 underflow interrupt
2	PWM2_INT_FINISH_EN	Set to 1 to enable PWM2 finish interrupt
1	PWM1_INT_UNDERFLOW_EN	Set to 1 to enable PWM1 underflow interrupt
0	PWM1_INT_FINISH_EN	Set to 1 to enable PWM1 finish interrupt

11006204 **PWM_INT_STATUS** **PWM Interrupt Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							PWM5_INT_UNDERFLOW_ST	PWM5_INT_FINISH_ST	PWM4_INT_UNDERFLOW_ST	PWM4_INT_FINISH_ST	PWM3_INT_UNDERFLOW_ST	PWM3_INT_FINISH_ST	PWM2_INT_UNDERFLOW_ST	PWM2_INT_FINISH_ST	PWM1_INT_UNDERFLOW_ST	PWM1_INT_FINISH_ST
Type							RU	RU	RU	RU	RU	RU	RU	RU	RU	RU
Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9	PWM5_INT_UNDERFLOW_ST	PWM5 underflow status
8	PWM5_INT_FINISH_ST	PWM5 finish status
7	PWM4_INT_UNDERFLOW_ST	PWM4 underflow status
6	PWM4_INT_FINISH_ST	PWM4 finish status
5	PWM3_INT_UNDERFLOW_ST	PWM3 underflow status
4	PWM3_INT_FINISH_ST	PWM3 finish status
3	PWM2_INT_UNDERFLOW_ST	PWM2 underflow status
2	PWM2_INT_FINISH_ST	PWM2 finish status
1	PWM1_INT_UNDERFLOW_ST	PWM1 underflow status
0	PWM1_INT_FINISH_ST	PWM1 finish status

11006208 **PWM_INT_ACK** **PWM Interrupt Acknowledge Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							PWM5_INT_UNDERFLOW_ACK	PWM5_INT_FINISH_ACK	PWM4_INT_UNDERFLOW_ACK	PWM4_INT_FINISH_ACK	PWM3_INT_UNDERFLOW_ACK	PWM3_INT_FINISH_ACK	PWM2_INT_UNDERFLOW_ACK	PWM2_INT_FINISH_ACK	PWM1_INT_UNDERFLOW_ACK	PWM1_INT_FINISH_ACK
Type							WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

11006208 PWM_INT_ACK PWM Interrupt Acknowledge Register 00000000

Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
9	PWM5_INT_UNDERFLOW_ACK	Set to 1 to clear PWM5 underflow interrupt
8	PWM5_INT_FINISH_ACK	Set to 1 to clear PWM5 finish interrupt
7	PWM4_INT_UNDERFLOW_ACK	Set to 1 to clear PWM4 underflow interrupt
6	PWM4_INT_FINISH_ACK	Set to 1 to clear PWM4 finish interrupt
5	PWM3_INT_UNDERFLOW_ACK	Set to 1 to clear PWM3 underflow interrupt
4	PWM3_INT_FINISH_ACK	Set to 1 to clear PWM3 finish interrupt
3	PWM2_INT_UNDERFLOW_ACK	Set to 1 to clear PWM2 underflow interrupt
2	PWM2_INT_FINISH_ACK	Set to 1 to clear PWM2 finish interrupt
1	PWM1_INT_UNDERFLOW_ACK	Set to 1 to clear PWM1 underflow interrupt
0	PWM1_INT_FINISH_ACK	Set to 1 to clear PWM1 finish interrupt

1100620C PWM_EN_STATUS PWM Enable Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												**PWM5_EN_ST**	**PWM4_EN_ST**	**PWM3_EN_ST**	**PWM2_EN_ST**	**PWM1_EN_ST**
Type												RO	RO	RO	RO	RO
Reset												0	0	0	0	0

Bit(s)	Name	Description
4	PWM5_EN_ST	PWM5 enabling status
3	PWM4_EN_ST	PWM4 enabling status
2	PWM3_EN_ST	PWM3 enabling status
1	PWM2_EN_ST	PWM2 enabling status
0	PWM1_EN_ST	PWM1 enabling status

11 SPI Interface Controller

11.1 Introduction

The SPI interface is a bit-serial, four-pin transmission protocol.

The SPI interface controller is a master responsible of the data transmission with the slave.

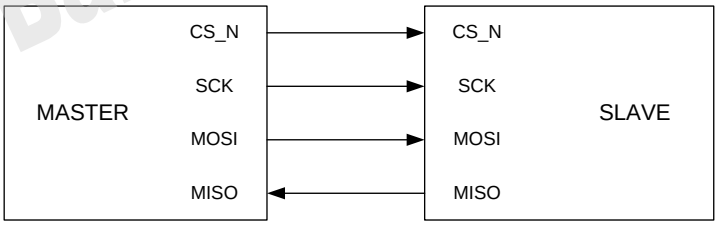


Figure 11-1: Pin connection between SPI master and SPI slave

11.2 Feature list

There are total 3 channel SPI controllers:

- Configurable CS_N setup time, hold time and idle time
- Programmable SCK high time and low time
- Configurable transmitting and receiving bit order
- Two configurable modes for the source of the data to be transmitted.
  1) In TX DMA mode, the SPI controller automatically fetches the transmitted data (to be put on the MOSI line) from memory;
  2) In TX FIFO mode, the data to be transmitted on the MOSI line are written to FIFO before the start of the transaction.
- Two configurable modes for destination of the data to be received.
  1) In RX DMA mode, the SPI controller automatically stores the received data (from MISO line) to memory;
  2) In RX FIFO mode, the received data keep being in RX FIFO of the SPI controller. The processor must read back the data by itself.
- Adjustable endian order from/to memory system
- Programmable byte length for transmission
- Unlimited length for transmission.
- Configurable option to control CS_N de-assert between byte transfers.

11.3 Block Diagram

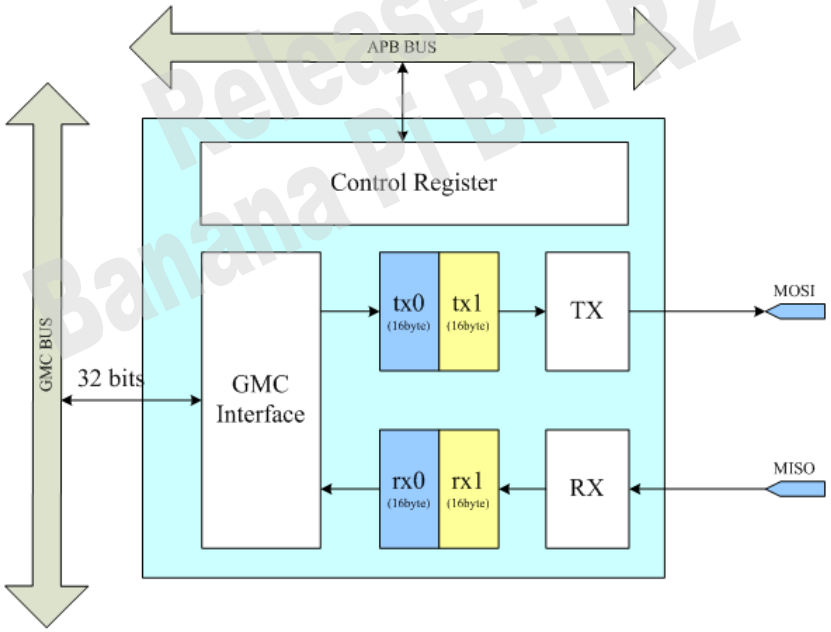


Figure 11-2: Block Diagram of SPI Controller

11.4 Register Definition

Module name: SPI0 base address: (+1100A000h)
Module name: SPI1 base address: (+11016000h)

Address	Name	Width	Register function
00000000	**SPI_CFG0**	32	SPI configuration 0 register
00000004	SPI_CFG1	32	SPI configuration 1 register
00000008	SPI_TX_SRC	32	SPI TX source address register
0000000C	SPI_RX_DST	32	SPI RX destination address register
00000010	SPI_TX_DATA	32	SPI TX data FIFO
00000014	SPI_RX_DATA	32	SPI RX data FIFO
00000018	SPI_CMD	32	SPI command register
0000001C	SPI_STATUS0	32	SPI status 0 register
00000020	SPI_STATUS1	32	SPI status 1 register

00000000 SPI_CFG0 SPI Configuration 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Mne	CS_SETUP_COUNT								CS_HOLD_COUNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mne	SCK_LOW_COUNT								SCK_HIGH_COUNT							

00000000 SPI_CFG0 SPI Configuration 0 Register 00000000

Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:24	**CS_SETUP_COUNT**	CS_SETUP_COUNT	Chip selection setup time = (CS_SETUP_COUNT+1)*CLK_PERIOD, where CLK_PERIOD is the cycle time of the clock the SPI engine adopts.
23:16	**CS_HOLD_COUNT**	CS_HOLD_COUNT	Chip selection hold time = (CS_HOLD_COUNT+1)*CLK_PERIOD
15:8	**SCK_LOW_COUNT**	SCK_LOW_COUNT	SCK clock low time = (SCK_LOW_COUNT+1)*CLK_PERIOD.
7:0	**SCK_HIGH_COUNT**	SCK_HIGH_COUNT	SCK clock high time = (SCK_HIGH_COUNT+1)*CLK_PERIOD

00000004 SPI_CFG1 SPI Configuration 1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**GET_TICK_DLY2**	**GET_TICK_DLY1**					**PACKET_LENGTH**									
Type	RW	RW					RW									
Reset	0	0					0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**PACKET_LOOP_CNT**								**CS_IDLE_COUNT**							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**GET_TICK_DLY2**	GET_TICK_DLY2	**If the speed of SPI is not fast enough, these two bits can help tolerate get_tick timing**
30	**GET_TICK_DLY1**	GET_TICK_DLY1	The timing range between get_tick and get_tick_dly1 is one cycle depending on the SPI system clock. The timing range between get_tick_dly1 and get_tick_dly2 is also one cycle depending on the SPI system clock. One cycle of SPI system clock is around 7.519 ns.us. Get_tick_dly1 can tolerate 8.138ns, and get_tick_dly2 can tolerate 15.038ns.
25:16	**PACKET_LENGTH**	PACKET_LENGTH	
15:8	**PACKET_LOOP_CNT**	PACKET_LOOP_CNT	The transmission on the SPI bus consists of units of bytes. Hence, PACKET_LENGTH[9:0] defines the number of bytes in one packet, and PACKET_LOOP_CNT[7:0] defines the number of packets within one transaction. The number of bytes in one packet = PACKET_LENGTH + 1. The number of packets in one transaction = PACKET_LOOP_CNT + 1. The total bytes of one transaction = (PACKET_LENGTH + 1)*(PACKET_LOOP_CNT + 1).
7:0	**CS_IDLE_COUNT**	CS_IDLE_COUNT	The chip selection idle time between consecutive transaction = (CS_HOLD_COUNT+1)*CLK_PERIOD.

00000008 SPI_TX_SRC SPI TX Source Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPI_TX_SRC[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPI_TX_SRC[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SPI_TX_SRC**	SPI_TX_SRC	If TX_DMA_EN is set, the data to be put on the MOSI line are kept in memory in advance, and the SPI controller will automatically read the data from memory. SPI_TX_SRC defines the memory address from which the SPI controller starts to read data.

0000000C SPI_RX_DST SPI RX Destination Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPI_RX_DST[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPI_RX_DST[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SPI_RX_DST**	SPI_RX_DST	If RX_DMA_EN is set, the received data from the MISO line will be moved to memory automatically by the SPI controller. SPI_RX_DST defines the memory address to which the SPI controller starts to store the data.

00000010 SPI_TX_DATA SPI TX Data FIFO 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPI_TX_DATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPI_TX_DATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SPI_TX_DATA**	SPI_TX_DATA	The depth of the TX FIFO is 32 bytes. Writing to this register will write 4 bytes to TX FIFO. The TX FIFO pointer will automatically move towards the next four bytes.

00000014 SPI_RX_DATA SPI RX Data FIFO 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPI_RX_DATA[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPI_RX_DATA[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SPI_RX_DATA**	SPI_RX_DATA	The depth of the RX FIFO is 32 bytes. Reading from this register will read 4 bytes from RX FIFO. The RX FIFO pointer will automatically move towards the next four bytes.

00000018 SPI_CMD SPI Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															PAUSE_IE	FINISH_IE
Type															RW	RW
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_ENDIAN	RX_ENDIAN	RXMSBF	TXMSBF	TX_DMA_EN	RX_DMA_EN	CPOL	CPHA			CS_DEASSERT_EN	PAUSE_EN		RST	RESUME	CMD_ACT
Type	RW	RW	RW	RW	RW	RW	RW	RW			RW	RW		RW	WO	WO
Reset	0	0	0	0	0	0	0	0			0	0		0	0	0

Bit(s)	Mnemonic	Name	Description
17	**PAUSE_IE**	PAUSE_IE	**Interrupt enabling bit of pause flag in SPI status register**
16	**FINISH_IE**	FINISH_IE	**Interrupt enabling bit of finish flag in SPI status register**
15	**TX_ENDIAN**	TX_ENDIAN	**Defines whether to reverse the endian order of the data DMA from memory** The default setting (0) is not to be reversed. Only supports DMA mode.
14	**RX_ENDIAN**	RX_ENDIAN	**Defines whether to reverse the endian order of the data DMA to memory** The default setting (0) is not to be reversed.
13	**RXMSBF**	RXMSBF	**Indicates the data received from MISO line is MSB first or not** Set RXMSBF to 1 for MSB first; otherwise set it to 0.
12	**TXMSBF**	TXMSBF	**Indicates the data sent on MOSI line is MSB first or not** Set TXMSBF to 1 for MSB first; otherwise set it to 0.
11	**TX_DMA_EN**	TX_DMA_EN	**DMA mode enabling bit of the data to be transmitted** The default setting (0) is not to be enabled.
10	**RX_DMA_EN**	RX_DMA_EN	**DMA mode enabling bit of the data received** The default setting (0) is not to be enable.
9	**CPOL**	CPOL	**Control bit of the SCK polarity** 0: CPOL = 0 1: CPOL = 1

Bit(s)	Mnemonic	Name	Description
8	**CPHA**	CPHA	**Defines the SPI Clock Format 0 or SPI Clock Format 1 during transmission** 0: CPHA = 0 1: CPHA = 1
5	**CS_DEASSERT_EN**	CS_DEASSERT_EN	**Enabling bit of the chip selection de-assertion mode** Set to 1 to enable this mode.
4	**PAUSE_EN**	PAUSE_EN	**Enabling bit of the pause mode** Set to 1 to enable this mode.
2	**RST**	RST	**Software reset bit** When this bit is 1, software reset will be active high. Default value: 0
1	**RESUME**	RESUME	**Used when controller is in PAUSE IDLE state** Writing 1 to this bit will trigger the SPI controller to resume transfer from PAUSE IDLE state.
0	**CMD_ACT**	CMD_ACT	**Command activate bit** Writing 1 to this bit will trigger the SPI controller to start the transaction.

0000001C SPI_STATUS0 SPI Status 0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															**PAUSE**	**FINISH**
Type															RC	RC
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1	**PAUSE**	PAUSE	**Interrupt status bit in pause mode** It will be set by the SPI controller when it completes the transaction, entering the PAUSE IDLE state.
0	**FINISH**	FINISH	**Interrupt status bit in non-pause mode** It will be set by the SPI controller when it completes the transaction, entering the IDLE state.

00000020 SPI_STATUS1 SPI Status 1 Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**IDLE**
Type																RU
Reset																1

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
0	**IDLE**	IDLE	**This status flag reflects the SPI controller is idle or not.** This bit is low active, i.e. 0 represents the SPI controller is busy now.

11.5 Programming Guide

Follow the steps below to perform SPI transmission:

1. Prepare the data in the memory with its start address to be the “source address”.
2. Set up the timing and protocol for the SPI transmission (see figure below for detailed setup parameters).
3. Fill the “destination address”, which is the start address that you would like to place the received data, and “source address”, which is the start address to place the data to be transmitted, into register SPI_RX_DST and SPI_TX_SRC, respectively.
4. Set up the CMD_ACT (bit0 of SPI_CMD) to start the transfer
5. Get the data received from the buffer prepared starting from “destination address”.

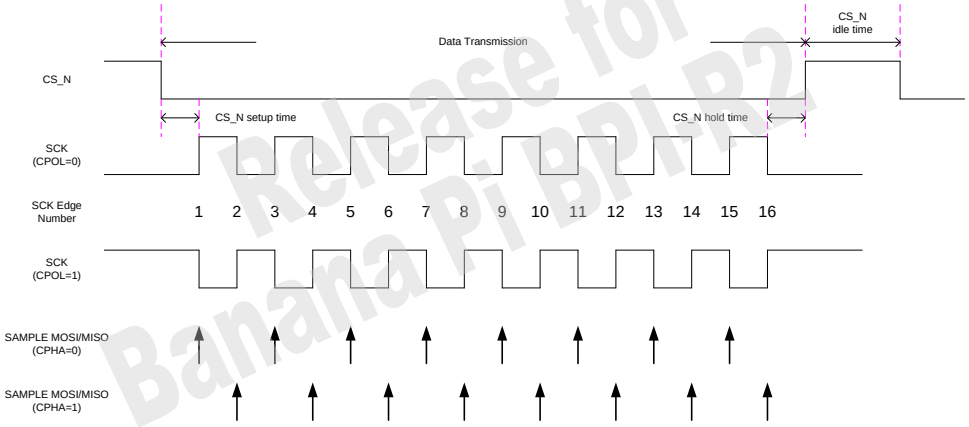


Figure 11-3: SPI transmission formats

Figure shows the waveform during the SPI transmission. The low active CS_N determines the start point and end point of one transaction. The CS_N setup time, hold time and idle time are also depicted.

CPOL defines the clock polarity in the transmission. Two types of polarity can be adopted, i.e. polarity 0 and polarity 1. It shows both of the clock polarity (CPOL) as examples.

CPHA defines the legal timing to sample MOSI and MISO. Two different methods can be adopted.

12 USB2.0

12.1 Introduction

The USB controller is configured for supporting 8 endpoints to receive packets and 8 endpoints to send packets except for endpoint 0. These endpoints can be individually configured in the software to handle either Bulk transfers, Interrupt transfers or Isochronous transfers. There are 8 DMA channels and the embedded RAM size is configurable size up to 8K bytes. The embedded RAM can be dynamically configured to each endpoint. As the host for point-to-point communications, the controller maintains a frame counter and automatically schedules SOF, Isochronous, Interrupt and Bulk transfers.

12.2 Feature list

- USB2.0 with OTG
- 8 Tx and 8 Rx end point (exclude EP0)
- 4 Tx and 4 Rx USBQ
- Battery Charge Configuration
  - support BC1.1 host mode for SDP/CDP detection
  - support BC1.1 device mode for SDP/CDP/DCP detection with PMIC

Table 12-1: USB20 Feature List

Feature list	Description
USB specifications	USB2.0 OTG
Enhanced feature	Generic Host QMU Generic Dev QMU
Endpoint	8 Tx 8 Rx EP0
DMA channel	8
Embedded RAM	Up to 8KB
UTMI+ interface	UTMI+ 16b
CPU slave interface	AHB asynchronous design
DMA master interface	AHB Busy Free Asynchronous Design
Base Address	32'h11200000

12.3 Block Diagram

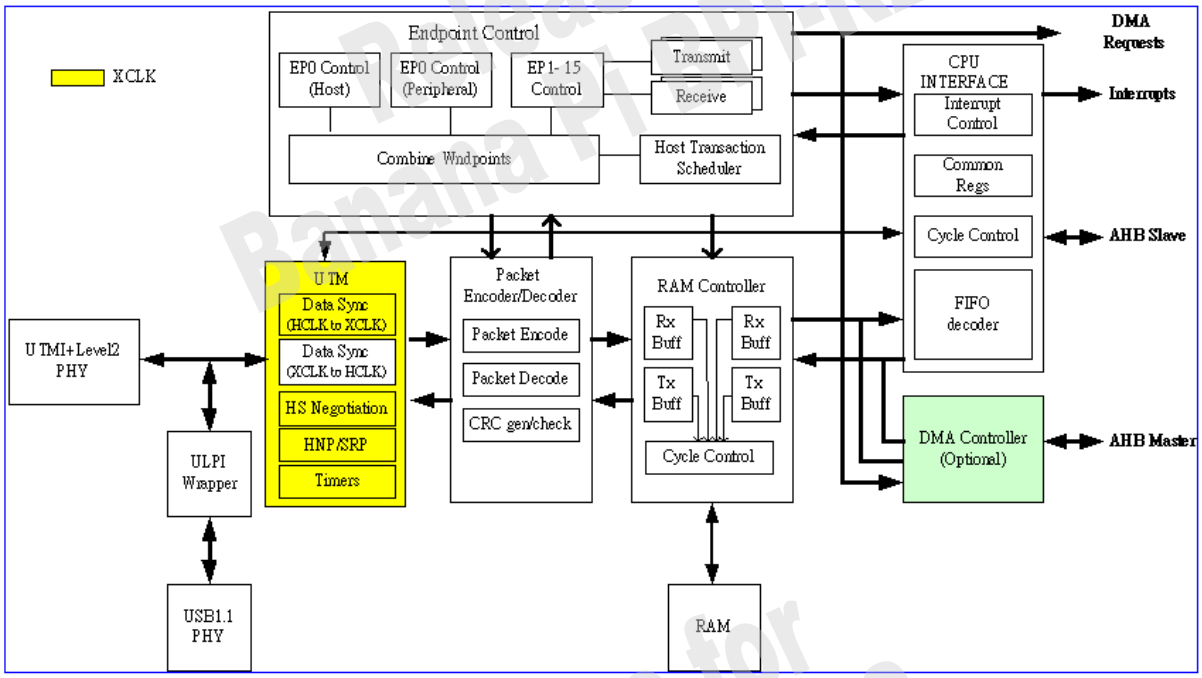


Figure 12-1: Block Diagram of USB2.0

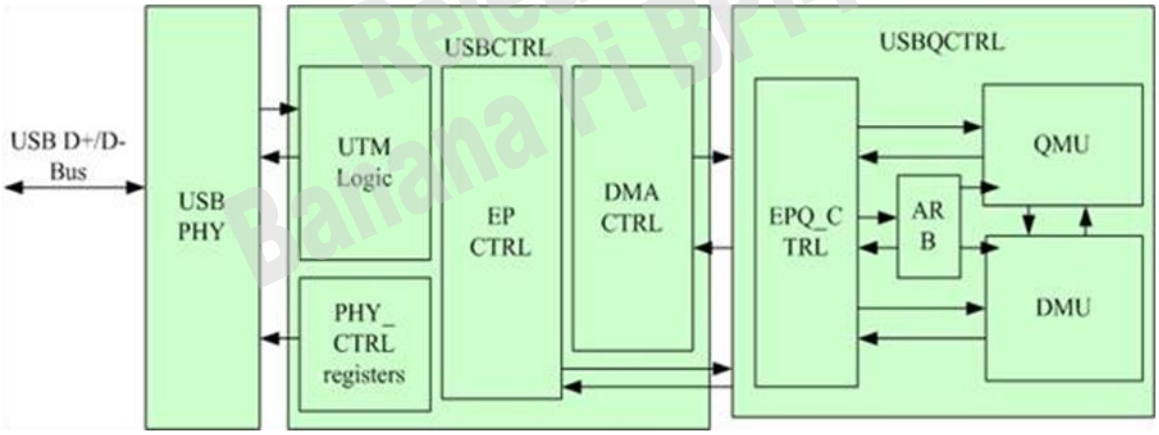


Figure 12-2: Block Diagram of USB2.0 QMU

12.4 USB MAC Register Definition

Registers accessed using byte manipulation are marked in blue columns. Byte accessing registers can be accessed using word manipulation. Word accessing registers cannot be accessed using the byte manipulation.

Register address	Register name	Manipulation	Acronym

		(Byte/Word)	
Common Registers			
USB + 0000h	Function address register	Byte	FADDR
USB + 0001h	Power management register	Byte	POWER
USB + 0002h	Tx interrupt status register	Byte	INTRTX
USB + 0004h	Rx interrupt status register	Byte	INTRRX
USB + 0006h	Tx interrupt enable register	Byte	INTRTXE
USB + 0008h	Rx interrupt enable register	Byte	INTRRXE
USB + 000Ah	Common USB interrupts register	Byte	INTRUSB
USB + 000Bh	Common USB interrupts enable register	Byte	INTRUSBE
USB + 000Ch	Frame number register	Byte	FRAME
USB + 000Eh	Endpoint selecting index register	Byte	INDEX
Indexed EndPoint CSR Region *n stands for endpoint number.* *For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint.* *MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0010h ~ USB + 001Fh	It maps to CSR EP0 ~ EPx depending on the INDEX register. For example, if INDEX is n, address 0010h ~ 001Fh are mapped to 0x(100+10*n)h ~ 0x(100+10*n+F)h.	Byte	Indexed CSR
USB + 0020h	USB endpoint 0 FIFO register	Byte	FIFO0
USB + 0020h +(n)*4 h	USB endpoint n FIFO register	Byte	FIFOn
OTG, Dynamic FIFO, Version Registers			
USB + 0060h	OTG device control register	Byte	DEVCTL
USB + 0061h	Power up counter register	Byte	PWRUPCNT
USB + 0062h	Tx FIFO size register	Byte	TXFIFOSZ
USB + 0063h	Rx FIFO size register	Byte	RXFIFOSZ
USB + 0064h	Tx FIFO address register	Byte	TXFIFOADD
USB + 0066h	Rx FIFO address register	Byte	RXFIFOADD
Hardware Configuration, Special Setting Registers			
USB + 007Ch	Time buffer available on HS transactions register	Byte	HS_EOF1
USB + 007Dh	Time buffer available on FS transactions register	Byte	FS_EOF1
USB + 007Eh	Time buffer available on LS transactions register	Byte	LS_EOF1
USB + 007Fh	Reset information register	Byte	RST_INFO
USB + 0080h	Rx data toggle set/status register	Word	RXTOG
USB + 0082h	Rx data toggle enable register	Word	RXTOGEN
USB + 0084h	Tx data toggle set/status register	Word	TXTOG
USB + 0086h	Tx data toggle enable register	Word	TXTOGEN

Level1 interrupt Control/Status registers			
USB + 00A0h	USB Level 1 interrupt status register	Byte	USB_L1INTS
USB + 00A4h	USB Level 1 interrupt unmask register	Byte	USB_L1INTM
USB + 00A8h	USB Level 1 interrupt polarity register	Byte	USB_L1INTP
USB + 00ACh	USB Level 1 interrupt control register	Byte	USB_L1INTC
Non-indexed EndPoint CSR Region *n stands for endpoint number.* *For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint.* *MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0102h	EP0 control status register	Byte	CSR0
USB + 0108h	EP0 received bytes register	Byte	COUNT0
USB + 010Bh	NAK limit register	Byte	NAKLIMT0
USB + 010Fh	Core configuration register	Byte	CONFIGDATA
USB + 0100h +(n)*10h	TXMAP register	Byte	TXMAP(n)
USB + 0102h +(n)*10h	Tx CSR register	Byte	TXCSR(n)
USB + 0104h +(n)*10h	RXMAP register	Byte	RXMAP(n)
USB + 0106h +(n)*10h	Rx CSR register	Byte	RXCSR(n)
USB + 0108h +(n)*10h	Rx Count register	Byte	RXCOUNT(n)
USB + 010Ah +(n)*10h	TxType register	Byte	TXTYPE(n)
USB + 010Bh +(n)*10h	TxInterval register	Byte	TXINTERVAL(n)
USB + 010Ch +(n)*10h	RxType register	Byte	RXTYPE(n)
USB + 010Dh +(n)*10h	RxInterval register	Byte	RXINTERVAL(n)
USB + 010Fh +(n)*10h	Configured FIFO size register	Byte	FIFOSIZE(n)
DMA Channels Control Registers *M stands for DMA channel number.* *For example, DMA channel 1's M = 1. Valid M = 1 ~ MaxDMAChannel.* *MaxDMAChannel is hardware configured and the maximum is 8.*			
USB + 0200h	DMA interrupt status register (word access only)	Word	DMA_INTR
USB + 0210h	DMA limiter register (word access only)	Word	DMA_LIMITER
USB + 0220h	DMA configuration register (word access only)	Word	DMA_CONFIG
USB + 0204h +(M-1)*10h	DMA channel M control register (word access only)	Word	DMA_CNTL_M

USB + 0208h +(M-1)*10h	DMA channel M address register (word access only)	Word	DMA_ADDR_M
USB + 020Ch +(M-1)*10h	DMA channel M byte count register (word access only)	Word	DMA_COUNT_M
EndPoint RX Packet Count Register *n stands for endpoint number. For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint. MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0300h +(n)*4h	EPn RxPktCount register	Word	EPnRXPKTCOUNT
Host/Hub Control Registers (Host mode only registers) *n stands for endpoint number. For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint. MaxEndPoint is hardware configured and maximum is 15.*			
USB + 0480h +8*n h	Transmit endpoint n function address	Word	TXFUNCADDR
USB + 0482h +8*n h	Transmit endpoint n hub/port address	Word	TXHUBADDR
USB + 0484h +8*n h	Receive endpoint n function address	Word	RXFUNCADDR
USB + 0486h +8*n h	Receive endpoint n hub/port address	Word	RXHUBADDR

12.4.1 Function Address Register

00 FADDR Function Address Register(Device mode only) 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										FUNCTION_ADDRESS						
Type										RW						
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6:0	FUNCTION_ADDRESS	FADDR is an 8-bit register that should be written with a 7-bit address of the peripheral part of a transaction. When the USB2.0 controller is used in the peripheral mode (DevCtl.bit2 = 0), the register should be written with the address received through a SET_ADDRESS command, which will then be used for decoding the function address in subsequent token packets. When the USB2.0 controller is used in the host mode (DevCtl.bit2 = 1). The function address is configured by TXFUNCADDR and RXFUNCADDR.

12.4.2 Power Management Register

12.4.2.1 Peripheral Mode

01 POWER_PERI Power Management Register 20

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									ISOUPDATE	SOFTCONN	HSENAB	HSMODE	RESET	RESUME	SUSPENDMODE	ENABLESUSPENDM

Type									RW	RW	RW	RU	RU	RW	RU	RW
Reset									0	0	1	0	0	0	0	0

Bit(s)	Name	Description
7	ISOUPDATE	When set by the CPU, the USB2.0 controller will wait for an SOF token from the time TxPktRdy is set before sending the packet. If an IN token is received before an SOF token, then a zero length data packet will be sent. Note: Only valid in the peripheral mode, and this bit only affects endpoints performing isochronous transfers.
6	SOFTCONN	If the Soft Connect/Disconnect feature is enabled, then the USB D+/D- lines is enabled when this bit is set by the CPU and tri-stated when this bit is cleared by the CPU. In the peripheral FS mode, clearing the Softcon bit may need execution latency until USB BUS SE0 is detected by HW. Execution Latency ~= 1ms, such as SOF Packet EOP or RESET. In the peripheral HS mode, clearing the Softcon bit still needs execution latency until USB BUS SE0 is detected by HW. Execution Latency ~= 1us, such as HS idle. Note: Only needed to be set in the peripheral Mode. For the host mode, this bit will be set if DEVCTL[0] session bit is set and this bit should also be cleared if the session bit is cleared by CPU.
5	HSENAB	When set by the CPU, the USB2.0 controller will negotiate for the high-speed mode when the device is reset by the hub. If not set, the device will only operate in the full-speed mode.
4	HSMODE	When set, this read-only bit indicates high-speed mode successfully negotiated during the USB reset. In the peripheral mode, it becomes valid when the USB reset is completed (as indicated by the USB reset interrupt). In the host Mode, it becomes valid when the Reset bit is cleared. Remains valid in the duration of the session. Note: Allowance is made for Tiny-J signaling in selecting the transfer speed.
3	RESET	Set when Reset signaling is present on the bus. Note: This bit can be read/written by the CPU in the host mode but read-only in the peripheral mode.
2	RESUME	Set by the CPU to generate Resume signaling in the suspend mode. The CPU clears this bit after 10ms (max. 15ms) to end the Resume signaling. In the host mode, this bit is automatically set when Resume signaling from the target is detected while the USB2.0 controller is suspended.
1	SUSPENDMODE	In the host mode, the CPU sets up this bit to enter the suspend mode. In the peripheral mode, set this bit to enter the suspend mode. Cleared when the CPU reads the interrupt register or sets up the Resume bit above.
0	ENABLESUSPENDM	Set by the CPU to enable SUSPENDM output

12.4.2.2 Host Mode

01 POWER_HOST Power Management Register 20

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name											HSENAB	HSMODE	RESET	RESUME	SUSPENDMODE	ENABLESUSPENDM
Type											RW	RU	RW	RW	A0	RW
Reset											1	0	0	0	0	0

Bit(s)	Name	Description
5	HSENAB	When set by the CPU, the USB2.0 controller will negotiate for the high-speed mode when the device is reset by the hub. If not set, the device will only operate in the full-speed mode.
4	HSMODE	When set, this read-only bit indicates high-speed mode successfully negotiated during

		the USB reset. In the peripheral mode, it becomes valid when the USB reset is completed (as indicated by the USB reset interrupt). In the host mode, it becomes valid when the Reset bit is cleared. Remains valid in the duration of the session. Note: Allowance is made for Tiny-J signaling in selecting the transfer speed.
3	RESET	Set when Reset signaling is present on the bus Note: This bit can be read/written by the CPU in the host mode but read-only in the peripheral mode.
2	RESUME	Set by the CPU to generate Resume signaling in the suspend mode. The CPU clears this bit after 10ms (max. 15ms) to end the Resume signaling. In the host mode, this bit is automatically set when the Resume signaling from the target is detected while the USB2.0 controller is suspended.
1	SUSPENDMODE	In the host mode, the CPU sets this bit to enter the suspend mode. In the peripheral mode, set this bit to enter the suspend mode. Cleared when the CPU reads the interrupt register or sets up the Resume bit above.
0	ENABLESUSPENDM	Set by the CPU to enable SUSPENDM output

The “**HS Enab**” bit can be used to disable high-speed operation. Normally the USB MAC will automatically negotiate for high-speed operation when it is reset. If this bit is cleared, however, the USB MAC will remain in the full-speed mode even when connected to a high-speed-capable USB.

The “**HS Mode**” bit can be used to determine whether the USB MAC is in the high-speed mode or full-speed mode. It will go high when the function successfully negotiates for high-speed operation during a USB reset.

The “**Reset**” bit can be used to determine when the Reset signaling is present on the USB. It will go high when the Reset signaling is detected and remain high until the bus reverts to an idle state.

The “**Resume**” bit is used to force the USB MAC to generate the Resume signaling on the USB to perform remote wake-up from the suspend mode. Once set high, it should be left high for approximately 10ms (at least 1ms and no more than 15ms) then cleared.

The “**Suspend Mode**“ bit is set either to make the USB MAC enter the suspend mode (host mode) or when the suspend mode is entered (peripheral mode). It will be cleared when the IntrUSB register is read (as a result of receiving a suspend interrupt). It will also be cleared if the suspend mode is left by setting the “Resume” bit to initiating a remote wake-up.

The “**Enable SuspendM**“ bit enables the SUSPENDM signal to indicate when the USB MAC is in the suspend Mode. With this bit set to ‘1’, the SUSPENDM signal will be low whenever the USB MAC is in the suspend mode and high at all other time.

12.4.3 Tx Interrupt Status Register

0002 INTRTX Tx Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8_TX	EP7_TX	EP6_TX	EP5_TX	EP4_TX	EP3_TX	EP2_TX	EP1_TX	EP0
Type								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
8	EP8_TX	T8 endpoint N interrupt event
7	EP7_TX	T7 endpoint N interrupt event
6	EP6_TX	T6 endpoint N interrupt event
5	EP5_TX	T5 endpoint N interrupt event
4	EP4_TX	T4 endpoint N interrupt event
3	EP3_TX	T3 endpoint N interrupt event
2	EP2_TX	T2 endpoint N interrupt event
1	EP1_TX	T1 endpoint N interrupt event
0	EP0	Endpoint0 interrupt event

TXENDPOINTS can be read from the EPINFO register. The undefined endpoints bits are reserved.

12.4.4 Rx Interrupt Status Register

0004 INTRRX Rx Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8_RX	EP7_RX	EP6_RX	EP5_RX	EP4_RX	EP3_RX	EP2_RX	EP1_RX	
Type								W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	
Reset								0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
8	EP8_RX	R8 endpoint N interrupt event
7	EP7_RX	R7 endpoint N interrupt event
6	EP6_RX	R6 endpoint N interrupt event
5	EP5_RX	R5 endpoint N interrupt event
4	EP4_RX	R4 endpoint N interrupt event
3	EP3_RX	R3 endpoint N interrupt event
2	EP2_RX	R2 endpoint N interrupt event
1	EP1_RX	R1 endpoint N interrupt event

RXENDPOINTS can be read from the EPINFO register. The undefined endpoints bits are reserved.

12.4.5 Tx Interrupt Enable Register

0006 INTRTXE Tx Interrupt Enable Register FFFF

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8_TXE	EP7_TXE	EP6_TXE	EP5_TXE	EP4_TXE	EP3_TXE	EP2_TXE	EP1_TXE	EP0_E
Type								RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset								1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
8	EP8_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
7	EP7_TXE	1'b0: Disable Tx endpoint N interrupt event

		1'b1: Enable Tx endpoint N interrupt event
6	EP6_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
5	EP5_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
4	EP4_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
3	EP3_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
2	EP2_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
1	EP1_TXE	1'b0: Disable Tx endpoint N interrupt event 1'b1: Enable Tx endpoint N interrupt event
0	EP0_E	1'b0: Disable Tx endpoint0 interrupt event 1'b1: Enable Tx endpoint0 interrupt event

INTRTXE provides interrupt enabling bits for the interrupts in INTRTX. Where a bit is set to 1, MC_NINT will be asserted on the corresponding interrupt in the INTRTX status register becoming set. Where a bit is set to 0, the interrupt in INTRTX status is still set but MC_NINT is not asserted.

TXENDPOINTS can be read from the EPINFO register. The undefined endpoints bits are reserved.

12.4.6 Rx Interrupt Enable Register

0008 INTRRXE Rx Interrupt Enable Register FFFE

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8_REXE	EP7_REXE	EP6_REXE	EP5_REXE	EP4_REXE	EP3_REXE	EP2_REXE	EP1_REXE	
Type								RW	RW	RW	RW	RW	RW	RW	RW	
Reset								1	1	1	1	1	1	1	1	

		Description
		1'b0: Disable Rx Endpoint N interrupt event 1'b1: Enable Rx Endpoint N interrupt event
		1'b0: Disable Rx Endpoint N interrupt event 1'b1: Enable Rx Endpoint N interrupt event
		1'b0: Disable Rx Endpoint N interrupt event 1'b1: Enable Rx Endpoint N interrupt event

1'b0: Disable Rx Endpoint N interrupt event
1'b1: Enable Rx Endpoint N interrupt event

1'b0: Disable Rx Endpoint N interrupt event
1'b1: Enable Rx Endpoint N interrupt event

1'b0: Disable Rx Endpoint N interrupt event
1'b1: Enable Rx Endpoint N interrupt event

1'b0: Disable Rx Endpoint N interrupt event
1'b1: Enable Rx Endpoint N interrupt event

1'b0: Disable Rx Endpoint N interrupt event
1'b1: Enable Rx Endpoint N interrupt event

INTRRXE provides interrupt enabling bits for the interrupts in INTRRX. Where a bit is set to 1, MC_NINT will be asserted on the corresponding interrupt in the INTRRX status register becoming set. Where a bit is set to 0, the interrupt in INTRRX status is still set but MC_NINT is not asserted.

RXENDPOINTS can be read from the EPINFO register. The undefined endpoints bits are reserved.

12.4.7 Common USB Interrupt Register

Common USB Interrupt Register

Description

Set when VBus drops below the VBus valid threshold during a session. Only valid when USB2.0 controller is 'A' device.

Set when Session Request signaling has been detected. Only valid when USB2.0 controller is 'A' device.

Set in the host mode when a device disconnection is detected. Set in the peripheral mode when a session ends. Valid at all transaction speeds.

Set when a device connection is detected. Only valid in the host mode. Valid at all transaction speeds.

Set when a new frame starts.

Set in the peripheral mode when the Reset signaling is detected on the bus.

Set in the host mode when babble is detected. Note: Only active after the first SOF is sent.

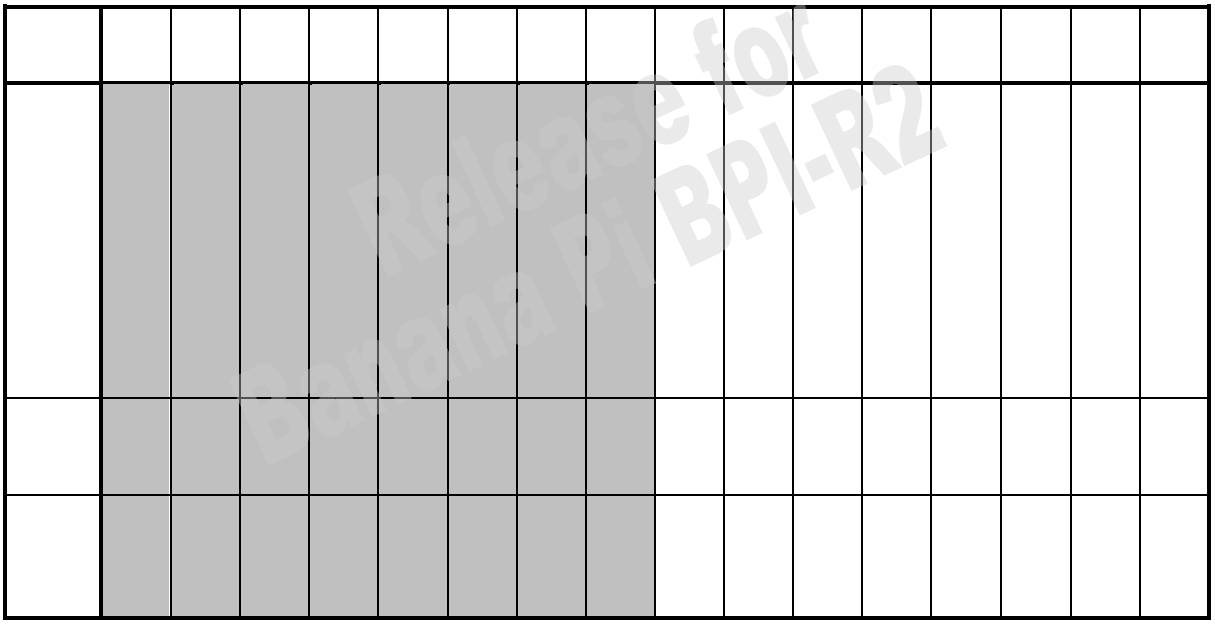
Set when the Resume signaling is detected on the bus while the USB2.0 controller is in the suspend

mode.

Set when the Suspend signaling is detected on the bus. Only valid in the peripheral mode.

12.4.8 Common USB Interrupt Enable Register

Common USB Interrupt Enable Register



Bit(s)	Name	Description
7	VBUSERROR_E	Enables VBusError interrupt
6	SESSREQ_E	Enables SessReq interrupt
5	DISCON_E	Enables Discon interrupt
4	CONN_E	Enables Conn interrupt
3	SOF_E	Enables SOF interrupt
2	RESET_BABLE_E	Enables Reset/Babble interrupt
1	RESEUM_E	Enables Resume interrupt
0	SUSPEND_E	Enables Suspend interrupt

INTRUSBE provides interrupt enabling bits for the interrupts in INTRUSB. Where a bit is set to 1, MC_NINT will be asserted on the corresponding interrupt in the INTRUSB status register becoming set. Where a bit is set to 0, the interrupt in INTRUSB status is still set but MC_NINT is not asserted

12.4.9 Frame Number Register

000C FRAME Frame Number Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						FRAME_NUMBER										
Type						RU										
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
10:0	FRAME_NUMBER	Frame is a 11-bit read-only register that holds the last received frame number.

FRAME NUBMER Frame is a 11-bit read-only register that holds the last received frame number.

12.4.10 Endpoint Selection Index Register

0E INDEX Endpoint Selection Index Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													SELECTED_ENDPOINT			
Type													RW			
Reset													0	0	0	0

Bit(s)	Name	Description
3:0	SELECTED_ENDPOINT	Each Tx endpoint and Rx endpoint has its own set of control/status registers located between USB+100h - USB+1FFh. In addition one set of Tx control/status and one set of Rx control/status registers appear at USB+010h - USB+01Fh. Index is a 4-bit register that determines which endpoint control/status registers are accessed. Before accessing an endpoint's control/status registers at USB+010h - USB+01Fh, the endpoint number should be written to the Index register to ensure that the correct control/status registers appear in the memory map.

12.4.11 USB Endpoint 0 FIFO Register

00000020 FIFO0 USB Endpoint 0 FIFO Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA[31:16]															
Type	Other															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA[15:0]															
Type	Other															
Reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit(s)	Name	Description
31:0	FIFO_DATA	The Endpoint FIFO registers provides 16 addresses for the CPU to access FIFOs for each endpoint. Writing to these addresses will load data into TxFIFO for the corresponding endpoint. Reading from these addresses will unload data from RxFIFO for the corresponding endpoint.

1. Note: Transfers to and from FIFOs may be 8-bit, 16-bit or 32-bit as required, and any combination of access is allowed provided the data accessed is contiguous. However, all the transfers associated with one packet must be of the same width so that the data are consistently byte-, word- or double-word-aligned. The last transfer may however contain fewer bytes than the previous transfers in order to complete an odd-byte or odd-word transfer. For DC/DTV projects, also refer to the RISC_SIZE register to complete the FIFO access.Depending on the size of FIFO and the expected maximum packet size, FIFO supports either single-packet or double-packet buffering. However, burst writing of multiple packets is not supported as flags need to be set after each packet is written.Following a STALL response or a Tx strike out error on endpoint, the associated FIFO is completely flushed.

The programmers does not neet to use debugging tools to monitor or read the FIFO region. The FIFO pointer will increase and cause unexpected errors in MAC state machine.

12.4.12 USB Endpoint n FIFO Register

00000024 FIFO1 USB Endpoint 1 FIFO Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA[31:16]															
Type	Other															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA[15:0]															
Type	Other															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA	The Endpoint FIFO register provides 16 addresses for the CPU to access the FIFO for each endpoint. Writing to these addresses loads data into TxFIFO for the corresponding endpoint. Reading from these addresses unloads data from RxFIFO for the corresponding endpoint. 1. Note:Transfers to and from FIFOs may be 8-bit, 16-bit or 32-bit as required, and any combination of access is allowed provided the data accessed is contiguous. However, all the transfers associated with one packet must be of the same width so that the data are consistently byte-, word- or double-word-aligned. The last transfer may however contain fewer bytes than the previous transfers in order to complete an odd-byte or odd-word transfer. For DC/DTV projects, also refer to the RISC_SIZE register to complete the FIFO access.Depending on the size of FIFO and the expected maximum packet size, FIFO supports either single-packet or double-packet buffering. However, burst writing of multiple packets is not supported as flags need to be set after each packet is written.Following a STALL response or a Tx strike out error on endpoint, the associated FIFO is completely flushed. The programmers does not need to use debugging tools to monitor or read the FIFO region. The FIFO pointer will increase and cause unexpected errors in the MAC state machine.

12.4.13 Device Control Register

60 DEVCTL Device Control Register NA

Bit	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
Na									B	F	L	VBUS		H	H	S
Ty									R	R	R	RU		R	C	C

Re									0	0	0	X	X	0	0	0

Bit(s)	Name	Description
7	B_DEVICE	This read-only bit indicates whether the USB2.0 controller is operating as the 'A' device or the 'B' device. Only valid while a session is in progress. Note: If the core is in Force_Host mode, this bit will indicate the state of the HOSTDISCON input signal from PHY. 1'b0: 'A' device 1'b1: 'B' device
6	FSDEV	This read-only bit is set when a full-speed or high-speed device has been detected being connected to the port. (High-speed devices are distinguished from full-speed by checking high-speed chirps when the device is reset.) Only valid in the host mode.
5	LSDEV	This read-only bit is set when a low-speed device has been detected being connected to the port. Only valid in the host mode.
4:3	VBUS	The read-only bits encode the current VBUS level as follows: (Only available with OTG function equipped, else the register value is undefined.) 2'b00: Below SessionEnd 2'b01: Above SessionEnd, below AValid 2'b10: Above AValid, below VBusValid 2'b11: About VBusValid
2	HOSTMODE	This read-only bit is set when the USB2.0 controller is acting as a host.
1	HOSTREQ	When set, the USB2.0 controller will initiate host negotiation when the suspend mode is entered. Cleared when host negotiation is completed. ('B' device only).
0	SESSION	When operating as an 'A' device, this bit is set or cleared by the CPU to start or end a session. When operating as a 'B' device, this bit is set/cleared by the USB2.0 controller when a session starts/ends. It is also set by the CPU to initiate the session request protocol. When the USB2.0 controller is in the suspend mode, the bit may be cleared by the CPU to perform a software disconnect. Note: Clearing this bit when the core is not suspended will result in undefined behavior.

12.4.14 PWRUPCNT Register

61 PWRUPCNT Power Up Counter Register 0F

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													PWRUPCNT			
Type													RW			
Reset													1	1	1	1

Bit(s)	Name	Description
3:0	PWRUPCNT	Power-up counter limit The power-up counter is used to count the K state duration during suspension, and when it is timed out, the resume interrupt will be issued. The register should be configured according to the AHB clock speed.

12.4.15 Tx FIFO Size Register (Dynamic FIFO Sizing Only)

62 TXFIFOSZ Tx FIFO Size Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												TXDPB	TXSZ			
Type												RW	RW			
Reset												0	0	0	0	0

	Name	Description
4	TXDPB	Defines whether the double-packet buffering is supported for TxFIFO 0: Support only single-packet buffering 1: Support double-packet buffering
3:0	TXSZ	Maximum packet size allowed (before any splitting within FIFO of Bulk/High-Bandwidth packets prior to transmission) If TxDPB = 0, FIFO will also be this size. If TxDPB = 1, FIFO will be twice this size. TxSZ[3:0]: Packet size (bytes) 4'b0000: 8 4'b0001: 16 4'b0010: 32 4'b0011: 64 4'b0100: 128 4'b0101: 256 4'b0110: 512 4'b0111: 1,024 4'b1000: 2,048 (single-packet buffering only) 4'b1001: 4,096 (single-packet buffering only) Others: No support

Only valid when configuring dynamic FIFO sizing.

The option of setting FIFO sizes dynamically only applies to Endpoints 1 ~.15. The Endpoint 0 FIFO has a fixed size(64 bytes) and a fixed location(start address 0)

It is the responsibility of the firmware (and the system designer) to ensure that all the Tx and Rx endpoints that are active in the current USB configuration have a block of RAM assigned exclusively to them which is at least as large as the maximum packet size set for the endpoint.

12.4.16 Rx FIFO Size Register (Dynamic FIFO Sizing Only)

(RX Rx FIFO Size Register

													RXSZ			

												RW				

	Na	Description
	RX	Defines whether double-packet buffering is supported for TxFIFO 0: Support only single-packet buffering 1: Support double-packet buffering
	RX	Maximum packet size allowed for (before any splitting within FIFO of Bulk/High-Bandwidth packets prior to transmission) If TxDPB = 0, FIFO will also be this size. If TxDPB = 1, FIFO will be twice this size. RxSZ[3:0]: Packet size (bytes) 4'b0000: 8 4'b0001: 16 4'b0010: 32 4'b0011: 64 4'b0100: 128 4'b0101: 256 4'b0110: 512 4'b0111: 1,024 4'b1000: 2,048 (single-packet buffering only) 4'b1001: 4,096 (single-packet buffering only) Others: No support

12.4.17 Tx FIFO Address Register (Dynamic FIFO Sizing Only)

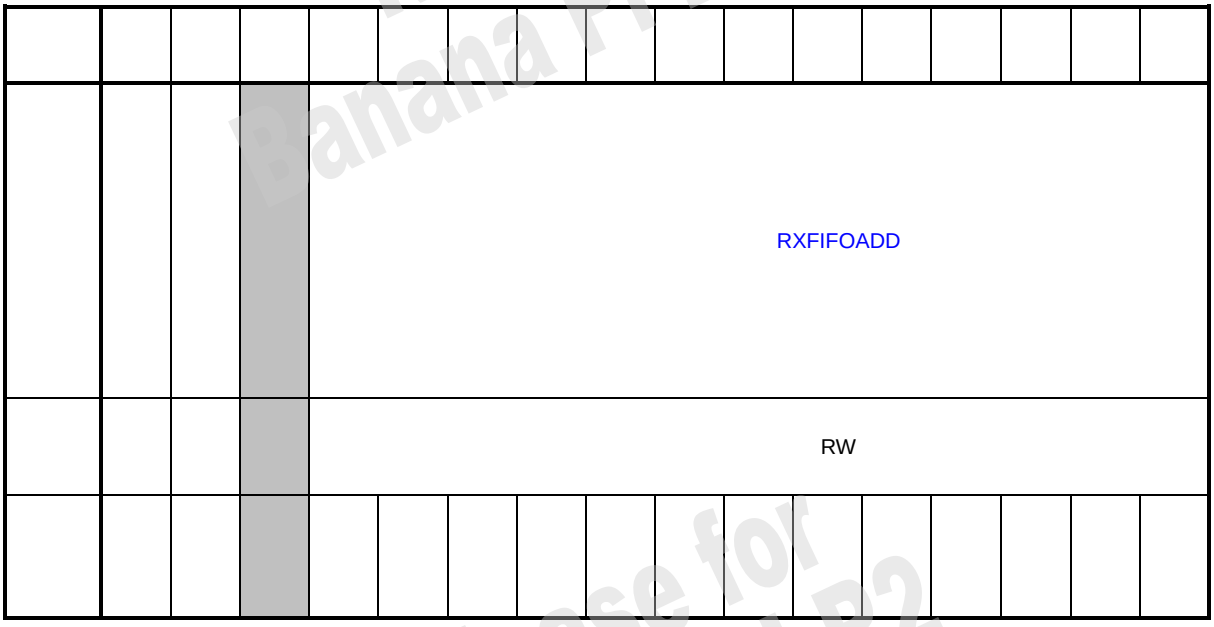
Tx FIFO Address Register

				TXFIFOADD												
				RW												

	Name	Description
:	TXFIFOADD	13-bit register which controls the start address of the selected Tx endpoint FIFO TxFIFOadd[12:0] Start address 13'h0000: 0000 13'h0001: 0008 13'h0002: 0010 13'h1FFF: FFF8

12.4.18 Rx FIFO Address Register (Dynamic FIFO Sizing Only)

(RX Rx FIFO Address Register



	Na	Description

Da	Enables data error interrupt Note: This bit is only valid when the endpoint is operating in the ISO mode.
Ov	Enables overrun interrupt Note: This bit is only valid when the endpoint is operating in the ISO mode.
RX	RxFIFOadd is a 13-bit register which controls the start address of the selected Rx endpoint FIFO. RxFIFOadd[12:0] Start address 13'h0000: 0000 13'h0001: 0008 13'h0002: 0010 13'h1FFF: FFF8

12.4.19 HS_EOF1 Register

7C HS_EOF1 Time buffer available on HS transaction Register 80

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									HS_EOF1							
Type									RW							
Reset									1	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	HS_EOF1	Sets up for high-speed transactions the time before EOF to stop beginning new transactions, in units of 133.3ns. (The default setting corresponds to 17.07us.). USB2.0 IP only.

12.4.20 FS_EOF1 Register

7D FS_EOF1 Time buffer available on FS transaction Register 77

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									FS_EOF1							
Type									RW							
Reset									0	1	1	1	0	1	1	1

Bit(s)	Name	Description
7:0	FS_EOF1	Sets up for full-speed transactions the time before EOF to stop beginning new transactions, in units of 533.3ns. (The default setting corresponds to 63.46us.) The default value will change to 8'hBE to meet 63.46us.

12.4.21 LS_EOF1 Register

7E LS_EOF1 Time buffer available on LS transaction Register 72

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									LS_EOF1							
Type									RW							
Reset									0	1	1	1	0	0	1	0

Bit(s)	Name	Description
7:0	LS_EOF1	Sets up for low-speed transactions the time before EOF to stop beginning new transactions, in units of 1.067us. (The default setting corresponds to 121.6us.). USB2.0 IP only. The default value will change to 8'hB6 to meet 121.6us.

12.4.22 RSTINFO Register

7F RST_INFO Reset Information Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									WTFSSE0				WTCHRP			
Type									RW				RW			
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:4	WTFSSE0	Signifies the SE0 signal duration before issuing the reset signal(for device only) Duration = 272.8*WTFSSE0 + 2.5 usec(This register will only be reset when the hardware is reset.)
3:0	WTCHRP	Sets up the delay to be applied from detecting reset to sending chirp K (for device only) Duration = 272.8*WTCHRP + 0.1 usec(This register will only be reset when the hardware is reset.)

12.4.23 Rx Data Toggle Set/Status Register

0080 RXTOG Rx Data Toggle Set/Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8RXTOG	EP7RXTOG	EP6RXTOG	EP5RXTOG	EP4RXTOG	EP3RXTOG	EP2RXTOG	EP1RXTOG	
Type								RW	RW	RW	RW	RW	RW	RW	RW	
Reset								0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
8	EP8RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable bit is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1

7	EP7RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
6	EP6RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
5	EP5RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
4	EP4RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
3	EP3RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
2	EP2RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1
1	EP1RXTOG	Receives logical Endpoint n data toggle bit set/status When read, these bits indicate the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n RX data toggle bit = 0 1'b1: Logical Endpoint n RX data toggle bit = 1

This register is word access.

12.4.24 Rx Data Toggle Enable Register

0082 RXTOGEN Rx Data Toggle Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8RXTOGEN	EP7RXTOGEN	EP6RXTOGEN	EP5RXTOGEN	EP4RXTOGEN	EP3RXTOGEN	EP2RXTOGEN	EP1RXTOGEN	
Type								RW	RW	RW	RW	RW	RW	RW	RW	
Reset								0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
8	EP8RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
7	EP7RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
6	EP6RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
5	EP5RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
4	EP4RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
3	EP3RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
2	EP2RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
1	EP1RXTOGEN	Receives logical Endpoint n data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG

This register is word access.

12.4.25 Tx Data Toggle Set/Status Register

0084 TXTOG Tx Data Toggle Set/Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name								EP8TXTOG	EP7TXTOG	EP6TXTOG	EP5TXTOG	EP4TXTOG	EP3TXTOG	EP2TXTOG	EP1TXTOG	
Type								RW	RW	RW	RW	RW	RW	RW	RW	
Reset								0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
8	EP8TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
7	EP7TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
6	EP6TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
5	EP5TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
4	EP4TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
3	EP3TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If the enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1
2	EP2TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. Ifthe enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1

1	EP1TXTOG	Transmits logical Endpoint n data toggle bit set/status When read, the bit indicates the current state of the Endpoint n data toggle. If the enable bit is high, the bit may be written with the required setting of the data toggle. If he enable is low, any value written will be ignored . Note: This register is word access. 1'b0: Logical Endpoint n TX data toggle bit = 0 1'b1: Logical Endpoint n TX data toggle bit = 1

This register is word access.

12.4.26 Tx Data Toggle Enable Register

0086 TXTOGEN Tx Data Toggle Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								EP8TXTOGEN	EP7TXTOGEN	EP6TXTOGEN	EP5TXTOGEN	EP4TXTOGEN	EP3TXTOGEN	EP2TXTOGEN	EP1TXTOGEN	
Type								RW	RW	RW	RW	RW	RW	RW	RW	
Reset								0	0	0	0	0	0	0	0	

Bit(s)	Name	Description
8	EP8TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
7	EP7TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
6	EP6TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
5	EP5TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
4	EP4TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
3	EP3TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
2	EP2TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling If the enable bit is set, the Endpoint n data toggle can be set. Note: This register is word access. 1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG 1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG
1	EP1TXTOGEN	Receives logical Endpoint 1 data toggle bit enabling

If the enable bit is set, the Endpoint n data toggle can be set.
Note: This register is word access.
1'b0: Forbid RISC writing EP n data toggle status with EP1RXTOG
1'b1: Allow RISC writing EP n data toggle status with EP1RXTOG

This register is word access.

12.4.27 USB Level 1 Interrupt Status Register(Word Access)

000000A0 USB_L1INTS USB Level 1 Interrupt Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					POWERDWN_INT_STATUS	DRVVBUS_INT_STATUS	IDDIG_INT_STATUS	VBUSVALID_INT_STATUS	DPDM_INT_STATUS	QHIF_INT_STATUS	QINT_STATUS	PSR_INT_STATUS	DMA_INT_STATUS	USBCOM_INT_STATUS	RX_INT_STATUS	TX_INT_STATUS
Type					RU	RU	RU	RU	RU	RU	RU	RU	RU	RU	RU	RU
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11	POWERDWN_INT_STATUS	Power -down interrupt status When controller is in the host suspend mode, VBus will be valid, DP asserted, and this bit set. When the controller is in the peripheral mode, Avalid will be set, DP asserted, and this bit set. When the controller is in the idle state, Avalid will be de-asserted, linestate in SE0, and this bit set.
10	DRVVBUS_INT_STATUS	DRVVBUS interrupt status This bit shows the interrupt trigger status of DRVVBUS. The trigger polarity is determined by DRVVBUS_INT_POL. This interrupt is used in USB OTG charge pump control.
9	IDDIG_INT_STATUS	IDDIG interrupt status This bit shows the interrupt trigger status of IDDIG. The trigger polarity is determined by IDDIG_INT_POL. This interrupt is used in USB OTG attachment.
8	VBUSVALID_INT_STATUS	VBUSVALID interrupt status This bit shows the interrupt trigger status of VBUSVALID. The trigger polarity is determined by VBUSVALID_INT_POL. This interrupt is used in USB attachment to host.
7	DPDM_INT_STATUS	DPDM interrupt status This bit shows the interrupt trigger status of DPDM. The trigger condition is whether DP or DM goes high. This interrupt is used in USB HOST mode to detect device attachment.
6	QHIF_INT_STATUS	USBQ HIF command interrupt status Only valid while WiMAX Q is available.
5	QINT_STATUS	USBQ interrupt status Only valid while USBQ is available.
4	PSR_INT_STATUS	Packet sequence recorder interrupt status
3	DMA_INT_STATUS	DMA interrupt status
2	USBCOM_INT_STATUS	USB common interrupt status
1	RX_INT_STATUS	Endpoint Rx interrupt status
0	TX_INT_STATUS	Endpoint Tx interrupt status

USB Level 1 interrupt status register. The USB interrupt will be fired only the mapped status and mask are all set. The status is read -only since SW need to clear it on the Level 2 register.

12.4.28 USB Level 1 Interrupt Mask Register(Word Access)

000000A4 USB_L1INTM USB Level 1 Interrupt Mask Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					POWERDWN_INT_UNMASK	DRVVBUS_INT_UNMASK	IDDIG_INT_UNMASK	VBUSVALID_INT_UNMASK	DPDM_INT_UNMASK	QHIF_INT_UNMASK	QINT_UNMASK	PSR_INT_UNMASK	DMA_INT_UNMASK	USBCOM_INT_UNMASK	RX_INT_UNMASK	TX_INT_UNMASK
Type					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11	POWERDWN_INT_UNMASK	Unmasks POWERDWN interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
10	DRVVBUS_INT_UNMASK	Unmasks DRVVBUS interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
9	IDDIG_INT_UNMASK	Unmasks IDDIG interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
8	VBUSVALID_INT_UNMASK	Unmasks VBUSVALID interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
7	DPDM_INT_UNMASK	Unmasks DPDM interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
6	QHIF_INT_UNMASK	Unmasks USBQ HIF command interrupt, Only valid while WiMAX Q is available. 1'b0: Mask interrupt 1'b1: Unmask interrupt
5	QINT_UNMASK	Unmasks USBQ interrupt, Only valid while USBQ is available 1'b0: Mask interrupt 1'b1: Unmask interrupt
4	PSR_INT_UNMASK	Unmasks packet sequence recorder interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
3	DMA_INT_UNMASK	Unmasks DMA interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
2	USBCOM_INT_UNMASK	Unmasks USB common interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
1	RX_INT_UNMASK	Unmasks Endpoint Rx interrupt 1'b0: Mask interrupt 1'b1: Unmask interrupt
0	TX_INT_UNMASK	Unmasks Endpoint Tx interrupt

1'b0: Mask interrupt
1'b1: Unmask interrupt

USB Level 1 interrupt unmask register. The interrupt will be fired only when unmask and status are both 1.

12.4.29 USB Level 1 Interrupt Polarity Register(Word Access)

000000A8 USB_L1INTP USB Level 1 Interrupt Polarity Register 00000200

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					POWERDWN_INT_POL	DRVVBUS_INT_POL	IDDIG_INT_POL	VBUSVALID_INT_POL								
Type					RW	RW	RW	RW								
Reset					0	0	1	0								

Bit(s)	Name	Description
11	POWERDWN_INT_POL	POWERDWN interrupt polarity 1'b0: Interrupt trigger when POWERDWN is 1. 1'b1: Interrupt trigger when POWERDWN is 0.
10	DRVVBUS_INT_POL	DRVVBUS interrupt polarity 1'b0: Interrupt trigger when DRVVBUS is 1. 1'b1: Interrupt trigger when DRVVBUS is 0.
9	IDDIG_INT_POL	IDDIG interrupt polarity 1'b0: Interrupt trigger when IDDIG is 1. 1'b1: Interrupt trigger when IDDIG is 0.
8	VBUSVALID_INT_POL	VBUSVALID interrupt polarity 1'b0: Interrupt trigger when VBUSVALID is 1. 1'b1: Interrupt trigger when VBUSVALID is 0.

USB Level 1 interrupt polarity register. The interrupt polarity is configured in this register.

12.4.30 USB Level 1 Interrupt Control Register(Word Access)

000000AC USB_L1INTC USB Level 1 Interrupt Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																USB_INT_SYNC
Type																RW
Reset																0

Bit(s)	Name	Description
0	USB_INT_SYNC	USB interrupt synchronization

1'b0: USB output interrupt is output directly.
1'b1: USB output interrupt is synchronized by MCU BUS clock registers.

USB Level 1 interrupt control register.

12.4.31 EP0 Control Status Register

12.4.31.1 Peripheral Mode

0102 CSR0_PERI EP0 Control Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								FLUSHFIFO	SERVICESETUPEDN	SERVICEDRXPKTRDY	SENDSTALL	SETUPEND	DATAEND	SENTSTALL	TXPKTRDY	RXPKTRDY
Type								A0	A0	A0	A0	RU	A0	RW	A0	RU
Reset								0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
8	FLUSHFIFO	The CPU writes a 1 to this bit to flush the next packet to be transmitted/read from the Endpoint 0 FIFO. Cleared automatically. The FIFO pointer is reset and the TxPktRdy/RxPktRdy bit (below) is cleared. May be set simultaneously with TxPktRdy to abort the packet that is currently being loaded into FIFO. Note: FlushFIFO should only be used when TxPktRdy/RxPktRdy is set. At other times, it may cause data corruption.
7	SERVICESETUPEDN	The CPU writes 1 to this bit to clear the SetupEnd bit. Cleared automatically.
6	SERVICEDRXPKTRDY	The CPU writes 1 to this bit to clear the RxPktRdy bit. Cleared automatically.
5	SENDSTALL	The CPU writes 1 to this bit to terminate the current transaction. The STALL handshake will be transmitted and then this bit will be cleared automatically. Note: FIFO should be flushed before SendStall is set.
4	SETUPEND	This bit will be set when a control transaction ends before the DataEnd bit has been set. An interrupt will be generated and FIFO flushed at this time. Cleared by the CPU by writing 1 to the ServicedSetupEnd bit.
3	DATAEND	The CPU sets up this bit: When setting up TxPktRdy for the last data packet. When clearing RxPktRdy after unloading the last data packet. When setting TxPktRdy for a zero length data packet. Cleared automatically
2	SENTSTALL	Set when a STALL handshake is transmitted. The CPU should clear this bit. Write 0 to clear.
1	TXPKTRDY	The CPU sets up this bit after loading a data packet into FIFO. Cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled).
0	RXPKTRDY	Set when a data packet has been received. An interrupt is generated when this bit is set. The CPU clears this bit by setting up the ServicedRxPktRdy bit.

12.4.31.2 Host Mode

0102 CSR0_HOST EP0 Control Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DISPING			FLUSHFIFO	NAKTIMEOUT	STATUSPKT	REQPKT	ERROR	SETUPPKT	RXSTALL	TXPKTRDY	RXPKTRDY
Type					RW			A0	A1	RW	RW	A1	A1	A1	A0	A1
Reset					0			0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11	DISPING	The CPU writes 1 to this bit to disable transmitting PING token. This bit can be set together with TXPKTRDY. To clear DISPING function, SW writes 0 after each control transfer is done. The DisPing function will be cleared when the next SETUP transaction is finished or OUT transaction receives handshake token with NAK/NYET/Timeout. Read "1" only in USB11 configuration. Token PING is not supported in FS mode. USB2.0 IP for TX only.
8	FLUSHFIFO	The CPU writes 1 to this bit to flush the next packet to be transmitted/read from the Endpoint 0 FIFO. Cleared automatically. The FIFO pointer is reset and the TxPktRdy/RxPktRdy bit (below) is cleared. May be set simultaneously with TxPktRdy to abort the packet that is currently being loaded into FIFO. Note: FlushFIFO should only be used when TxPktRdy/RxPktRdy is set. At other times, it may cause data corruption.
7	NAKTIMEOUT	This bit will be set when Endpoint 0 is halted following the receipt of NAK responses for longer than the time set by the NAKLimit0 register. The CPU clears this bit to allow the endpoint to continue. Write 0 to clear.
6	STATUSPKT	The CPU sets up this bit at the same time as the TxPktRdy or ReqPkt bit is set, to perform a status stage transaction. Setting up this bit ensures that the data toggle is set to 1 so that a DATA1 packet is used for the status stage transaction.
5	REQPKT	The CPU writes 1 to this bit to request an IN transaction. Cleared when RxPktRdy is set.
4	ERROR	This bit will be set when three attempts have been made to perform a transaction with no response from the peripheral. Cleared by the CPU. An interrupt is generated when this bit is set. Write 0 to clear.
3	SETUPPKT	The CPU sets up this bit, at the same time as the TxPktRdy bit is set, to send a SETUP token instead of an OUT token for the transaction. This bit will be automatically clear by HW when the setup packet is transmitted. Note: Setting up this bit will also clear DataToggle.
2	RXSTALL	This bit will be set when a STALL handshake is received. The CPU should clear this bit. Write 0 to clear.
1	TXPKTRDY	The CPU sets up this bit after loading a data packet into the FIFO. Cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled)
0	RXPKTRDY	This bit is set when a data packet has been received. An interrupt is generated when this bit is set. The CPU clears this bit by setting up the ServicedRxPktRdy bit. Write 0 to clear.

12.4.32 EP0 Received bytes Register

0108 COUNT0 EP0 Received bytes Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										EP0_RX_COUNT						
Type										RU						
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6:0	EP0_RX_COUNT	COUNT0 is a 7-bit read-only register that indicates the number of received data bytes in the Endpoint 0 FIFO. The value returned changes as the contents of FIFO change and is only valid while RxPktRdy (IDXEPR0.CSR0.bit0) is set.

12.4.33 NAK Limit Register

10B NAKLIMT0 NAK Limit Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												NAKLIMIT0				
Type												RW				
Reset												0	0	0	0	0

Bit(s)	Name	Description
4:0	NAKLIMIT0	(For host mode only) NAKLimit0 is a 5-bit register that sets up the number of frames/microframes (high-speed transfers) after which Endpoint 0 should timeout on receiving a stream of NAK responses. (Equivalent settings for other endpoints can be made through their TxInterval and RxInterval registers.). The number of frames/microframes selected is 2(m-1) (where m is the value set in the register, valid values 2 - 16). If the host receives NAK responses from the target for more frames than the number represented by the limit set in this register, the endpoint will be halted. Note: Value of 0 or 1 disables the NAK timeout function.

12.4.34 CONFIGDATA

10F CONFIGDATA Core Configuration Register 1F

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									MPRXE	MPTXE	BIGENDIAN	HBRXE	HBTXE	DYNFIFOSIZING	SOFTCONE	UTMIDATAWIDTH
Type									RO	RO	RO	RO	RO	RO	RO	RO
Reset									0	0	0	1	1	1	1	1

Bit(s)	Name	Description
7	MPRXE	When set to 1, automatic amalgamation of bulk packets is selected.
6	MPTXE	When set to 1, automatic splitting of bulk packets is selected.
5	BIGENDIAN	Set to 1 indicates big-endian ordering is selected.
4	HBRXE	Set to 1 indicates High-bandwidth Rx ISO endpoint support selected.
3	HBTXE	Set to 1 indicates High-bandwidth Tx ISO endpoint support selected.
2	DYNFIFOSIZING	Set to 1 indicates dynamic FIFO sizing option is selected.
1	SOFTCONE	Set to 1 indicates soft connection/disconnection option is selected.
0	UTMIDATAWIDTH	Indicates selected UTMI+ data width 1'b0: 8 bits 1'b1: 16 bits

12.4.35 TXMAP Register

0110 TX1MAP TX1MAP Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				M1		MAXIMUM_PAYLOAD_TRANSACTION										
Type				RW		RW										
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:11	M1	Maxmum payload size for indexed TX endpoint , M1 packet multiplier m maximum payload transaction register
10:0	MAXIMUM_PAYLOAD_TRANSACTION	The TxMaxP register defines the maximum amount of data that can be transferred through the selected Tx endpoint in asingle operation. There is a TxMaxP register for each Tx endpoint (except for Endpoint 0).Bits 10:0 define (in bytes) the maximum payload transmitted in a single transaction. The value set can be up to 1024 bytes but is

subject to the constraints placed by the USB specification on packet sizes for Bulk, Interrupt and Isochronous transfers in Fullspeed and High speed operation.

Where the option of High-bandwidth Isochronous endpoints or of packet splitting on Bulk endpoints has been taken when the core is configured, the register includes either 2 or 5 further bits that define a multiplier m which is equal to one more than the value recorded. In the case of Bulk endpoints with the packet splitting option enabled, the multiplier m can be up to 32 and defines the maximum number of the USB packets (i.e. packets for transmission over the USB) of the specified payload into which a single data packet placed in FIFO should be split, prior to transfer. (If the packet splitting option is not enabled, bit15-13 will not be implemented and bit12-11(if included) will be ignored.) The data packet is required to be an exact multiple of the payload specified by bits 10:0, which is itself required to be either 8, 16, 32, 64 or (in the case of High Speed transfers) 512 bytes. For Isochronous endpoints operating in high-speed mode and with the high-bandwidth option enabled, m may only be either 2 or 3 (corresponding to bit 11 set or bit 12 set, respectively) and it specifies the maximum number of such transactions that can take place in a single microframe. If either bit 11 or bit 12 is non-zero, the USB2.0 controller will automatically split any data packet written to FIFO into up to 2 or 3 USB packets, each containing the specified payload (or less). The maximum payload for each transaction is 1,024 bytes, so this allows up to 3072 bytes to be transmitted in each microframe. (For Isochronous/Interrupt transfers in full-speed mode, bits 11 and 12 are ignored.) The value written to bits 10:0 (multiplied by m in the case of high-bandwidth Isochronous transfers) must match the value given in the wMaxPacketSize field of the standard endpoint descriptor for the associated endpoint (see USB Specification Revision 2.0, Chapter 9). A mismatch can cause unexpected results.The total amount of data represented by the value written to this register (specified payload * m) must not exceed the FIFO size for the Tx endpoint and should not exceed half the FIFO size if double-buffering is required.If this register is changed after packets have been sent from the endpoint, the Tx endpoint FIFO should be completely flushed (using the FlushFIFO bit in TxCSR) after writing the new value to this register.

12.4.36 Tx CSR Register

12.4.36.1 Peripheral Mode

0112 TX1CSR_PERI Tx1 CSR Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUTOSET	ISO		DMAREQEN	FRCDATATOG	DMAREQMODE		SETTXPKTRDY_TWICE	INCOMPTX	CLRDATATOG	SENTSTALL	SENDSTALL	FLUSHFIFO	UNDERRUN	FIFONOTEMPTY	TXPKTRDY
Type	RW	RW		RW	RW	RW		A1	A1	A0	A1	RW	A0	A1	RU	A0
Reset	0	0		0	0	0		0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15	AUTOSET	If the CPU sets up this bit, TxPktRdy will be automatically set when data of the maximum packet size (value in TxMaxP) are loaded into TxFIFO. If a packet of smaller than the maximum packet size is loaded, TxPktRdy will have to be set manually.
14	ISO	The CPU sets up this bit to enable the Tx endpoint for Isochronous transfers and clears it to enable the Tx endpoint for Bulk or Interrupt transfers. Note: This bit only takes effect in the peripheral mode. In the host mode, it always returns to 0.
12	DMAREQEN	The CPU sets up this bit to enable the DMA request for the Tx endpoint.
11	FRCDATATOG	The CPU sets up this bit to force the endpoint data toggle to switch and the data packet to be cleared from the FIFO, regardless of whether an ACK is received. This can be used by Interrupt Tx endpoints that are used to communicate rate feedback for Isochronous endpoints.
10	DMAREQMODE	The CPU sets up this bit to select DMA request mode 1 and clears it to select DMA request mode 0.Note: This bit must not be cleared either before or in the same cycle as the DMAReqEn bit is cleared.

8	SETTXPKTRDY_TWICE	Indicates TxPktRdy had been set while it is 1'b1 already. Write 0 to clear.
7	INCOMPTX	When the endpoint is being used for high-bandwidth Isochronous/Interrupt transfers, this bit is set to indicate where a large packet has been split into 2 or 3 packets for transmission but insufficient IN tokens have been received to send all the parts. Note: In anything other than a high-bandwidth transfer, this bit will always return to 0. Write 0 to clear.
6	CLRDATATOG	The CPU writes 1 to this bit to reset the endpoint data toggle to 0.
5	SENTSTALL	This bit is set when a STALL handshake is transmitted. FIFO is flushed and Tx interrupt generated if enabled and the TxPktRdy bit is cleared. The CPU clears this bit. Write 0 to clear.
4	SENDSTALL	The CPU writes 1 to this bit to issue a STALL handshake to an IN token. The CPU clears this bit to terminate the stall condition. Note: This bit has no effect where the endpoint is being used for Isochronous transfers. Otherwise, CPU should wait SENTSTALL interrupt generated before clearning the SENDSTALL bit.
3	FLUSHFIFO	The CPU writes 1 to this bit to flush the latest packet from the endpoint TxFIFO. The FIFO pointer is reset, the TxPktRdy bit is cleared and an interrupt is generated. May be set simultaneously with TxPktRdy to abort the packet that is currently loaded into FIFO. Note: FlushFIFO should only be used when TxPktRdy is set. At other times, it may cause data corruption. If FIFO is double-buffered, FlushFIFO may need to be set twice to ompletely clear FIFO.
2	UNDERRUN	The USB sets up this bit if an IN token is received when the TxPktRdy bit not set. The CPU clears this bit (writing 0 to it).
1	FIFONOTEMPTY	The USB sets up this bit when there is at least 1 packet in TxFIFO. This bit will assert automatically when TXPKTRDY is set by CPU and de-assert when CPU flushes FIFO or sends a STALL packet.
0	TXPKTRDY	The CPU sets up this bit after loading a data packet into FIFO. Cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled). TxPktRdy is also automatically cleared (interrupt is generated) prior to loading a second packet into a double-buffered FIFO.

12.4.36.2 Host Mode

0112 TX1CSR_HOST Tx1 CSR Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUTOSET			DMAREQEN	FRCDATATOG	DMAREQMODE		SETTXPKTRDY_TWICE	NAKTIMEOUT_INCOMPTX	CLRDATAOG	RXSTALL		FLUSHFIFO	ERROR	FIFONOTEMPTY	TXPKTRDY
Type	RW			RW	RW	RW		A1	A1	A0	A1		A0	A1	RU	A0
Reset	0			0	0	0		0	0	0	0		0	0	0	0

Bit(s)	Name	Description
15	AUTOSET	If the CPU sets up this bit, TxPktRdy will be automatically set when data of the maximum packet size (value in TxMaxP) are loaded into TxFIFO. If a packet of smaller than the maximum packet size is loaded, then TxPktRdy will have to be set manually.
12	DMAREQEN	The CPU sets up this bit to enable the DMA request for the Tx endpoint.
11	FRCDATATOG	The CPU sets up this bit to force the endpoint data toggle to switch and the data packet to be cleared from FIFO, regardless of whether an ACK is received. This can be used by Interrupt Tx endpoints that are used to communicate rate feedback for Isochronous endpoints.
10	DMAREQMODE	The CPU sets up this bit to select DMA request mode 1 and clears it to select DMA request mode 0.Note: This bit must not be cleared either before or in the same cycle as the DMAReqEn bit is cleared.
8	SETTXPKTRDY_TWICE	This bit indicates TxPktRdy has been set while it is 1'b1 already. Write 0 to clear.
7	NAKTIMEOUT_INCOMPTX	Bulk endpoints only: This bit will be set when the Tx endpoint is halted following the receipt of NAK responses for longer than the time set as the NAK Limit by the TxInterval register. The CPU clears this bit to allow the endpoint to continue.Write 0 to clear.

6	CLRDATAOG	The CPU writes 1 to this bit to reset the endpoint data toggle to 0.
5	RXSTALL	This bit is set when a STALL handshake is received. When this bit is set, any DMA request that is in progress is topped, FIFO is completely flushed, Tx interrupt is generated if enabled and the TxPktRdy bit is cleared (see below). The CPU clears this bit. Write 0 to clear.
3	FLUSHFIFO	The CPU writes 1 to this bit to flush the latest packet from the endpoint TxFIFO. The FIFO pointer is reset, the TxPktRdy bit is cleared and an interrupt is generated. May be set simultaneously with TxPktRdy to abort the packet that is currently loaded into FIFO. Note: FlushFIFO should only be used when TxPktRdy is set. At other times, it may cause data corruption. If FIFO is double-buffered, FlushFIFO may need to be set twice to ompletely clear FIFO.
2	ERROR	The USB sets up this bit when 3 attempts have been made to send a packet and no handshake packet has been received. When the bit is set, an interrupt will be generated, TxPktRdy cleared and FIFO completely flushed. The CPU clears this bit. Valid only when the endpoint is operating in Bulk or Interrupt mode. Write 0 to clear.
1	FIFONOTEMPTY	The USB sets up this bit when there is at least 1 packet in TxFIFO. This bit will assert automatically when TXPKTRDY is set by CPU and de-assert when CPU flush FIFO or send a STALL packet.
0	TXPKTRDY	The CPU sets up this bit after loading a data packet into FIFO. It is cleared automatically when a data packet has been transmitted. An interrupt is also generated at this point (if enabled). TxPktRdy is also automatically cleared (interrupt is generated) prior to loading a second packet into a double-buffered FIFO.

12.4.37 RXMAP Register

0114 RX1MAP RX1MAP Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				M1		MAXIMUM_PAYLOAD_TRANSACTION										
Type				RW		RW										
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
12:11	M1	Maxmum payload size for indexed RX endpoint , M1 packet multiplier m
10:0	MAXIMUM_PAYLOAD_TRANSACTION	The RxMaxP register defines the maximum amount of data that can be transferred through the selected Rx endpoint in a single operation. There is a RxMaxP register for each Rx endpoint (except for Endpoint 0).Bits 10:0 define (in bytes) the maximum payload transmitted in a single transaction. The value set can be up to 1,024 bytes but is subject to the constraints placed by the USB specification on packet sizes for Bulk, Interrupt and Isochronous transfers in full-speed and high-speed operations. Where the option of high-bandwidth Isochronous endpoints or of combining Bulk packets has been taken when the core is configured, the register includes either 2 or 5 further bits that define a multiplier m which is equal to one more than the value recorded. For Bulk endpoints with the packet combining option enabled, the multiplier m can be up to 32 and defines the number of USB packets of the specified payload which are to be combined into a single data packet within the FIFO. (If the packet splitting option is not enabled, bit15-bit13 will not be implemented and bit12-bit11 (if included) will be ignored.) For Isochronous endpoints operating in high-speed mode and with the high-bandwidth option enabled, m may only be either 2 or 3 (corresponding to bit 11 set or bit 12 set, respectively) and it specifies the maximum number of such transactions that can take place in a single microframe. If either bit 11 or bit 12 is non-zero, the USB2.0 controller will automatically combine the separate USB packets received in any microframe into a single packet within the Rx FIFO. The maximum payload for each transaction is 1024 bytes, so this allows up to 3072 bytes to be received in each microframe. (For Isochronous/Interrupt transfers in full-speed mode or if high-bandwidth is not enabled, bits 11 and 12 will be ignored.) The value written to bits 10:0 (multiplied by m in the case of high-bandwidth Isochronous transfers) must match the value given in the wMaxPacketSize field of the standard endpoint descriptor for the associated endpoint

(see USB Specification Revision 2.0, Chapter 9). A mismatch could cause unexpected results.
The total amount of data represented by the value written to this register (specified payload * m) must not exceed the FIFO size for the OUT endpoint, and should not exceed half the FIFO size if double-buffering is required.

12.4.38 RX CSR Register

12.4.38.1 Peripheral Mode

0116 RX1CSR_PERI RX1 CSR Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUTOCLEAR	ISO	DMAREQEN	DISNYET_PIDERR	DMAREQMODE		KEEPERRSTATUS	INCOMPRX	CLRDTATOG	SENTSTALL	SENDSTALL	FLUSHFIFO	DATAERR	OVERRUN	FIFOFULL	RXPKTRDY
Type	RW	RW	RW	RW	RW		RW	A1	A0	A1	RW	A0	RU	A1	RU	A1
Reset	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15	AUTOCLEAR	If the CPU sets up this bit then the RxPktRdy bit will be automatically cleared when a packet of RxMaxP bytes has been unloaded from the RxFIFO. When packets of smaller than the maximum packet size are unloaded, RxPktRdy will have to be cleared manually. Note: The maximum packet size-3,-2,-1 is just handled like the maximum packet size which is auto cleared by hardware.
14	ISO	The CPU sets up this bit to enable the Rx endpoint for Isochronous transfers and clears it to enable the Rx endpoint for Bulk/Interrupt transfers.
13	DMAREQEN	The CPU sets up this bit to enable the DMA request for the Rx endpoint.
12	DISNYET_PIDERR	The CPU sets up this bit to disable the sending of NYET handshakes. When set, all successfully received Rx packets are ACK'd including at the point at which the RxFIFO becomes full. Note: This bit only has any effect in High-speed mode, in which mode it should be set for all interrupt endpoint. This bit is set when there is a PID error in the received packet. It is cleared when RxPktRdy is cleared or write 0 to clear.
11	DMAREQMODE	The CPU sets up this bit to select DMA request mode 1 and clears it to select DMA request mode 0. DMA request mode 1: Rx endpoint interrupt is generated only when DMA request mode 1 and receiving a short packet. RxDMAReq is generated when receiving a Max-Packet-size packet.DMA request mode 0: No Rx endpoint interrupt. RxDMAReq is generated when RxPktRdy is set.
9	KEEPERRSTATUS	This bit is used when endpoint working with USBQ and in ISOCHRONOUS mode. When this bit is set, the isochronous error, PIDERROR, INCOMPRX and DATAERROR will be kept and only cleared by SW.
8	INCOMPRX	This bit is set in a Isochronous transfer if the packet in the RxFIFO is incomplete because parts of the data are not received. Cleared when RxPktRdy is cleared or write 0 to clear it. Note: In anything other than a high-bandwidth transfer, this bit will always return to 0. Write 0 to clear.
7	CLRDTATOG	The CPU writes 1 to this bit to reset the endpoint data toggle to 0.
6	SENTSTALL	This bit is set when a STALL handshake is transmitted. The CPU clears this bit. An interrupt is generated when the bit is set. Write 0 to clear.
5	SENDSTALL	The CPU writes 1 to this bit to issue a STALL handshake. The CPU clears this bit to terminate the stall condition. Note: This bit has no effect where the endpoint is used for ISO transfers.
4	FLUSHFIFO	The CPU writes 1 to this bit to flush the next packet to be read from the endpoint RxFIFO. The RxFIFO pointer is reset and the RxPktRdy bit is cleared. Note: FlushFIFO should only be used when RxPktRdy is set. At other times, it may cause data corruption. If RxFIFO is double buffered, FlushFIFO may need to be set twice to completely clear

		RxFIFO.
3	DATAERR	This bit is set when RxPktRdy is set if the data packet has a CRC or bit-stuff error. Cleared when RxPktRdy is cleared. Note: This bit is only valid when the endpoint is operating in the ISO mode. In the bulk mode, it always returns to 0.
2	OVERRUN	This bit is set if an OUT packet cannot be loaded into RxFIFO. The CPU clears this bit (writing 0 to it). Note: This bit is only valid when the endpoint is operating in the ISO mode. In the bulk mode, it always returns to 0. The new incoming packet will not be written to RxFIFO.
1	FIFOFULL	This bit is set when no more packets can be loaded into RxFIFO.
0	RXPKTRDY	This bit is set when a data packet has been received (to RxFIFO). The CPU clears this bit when the packet has been unloaded from RxFIFO. An interrupt will be generated when the bit is set. Write 0 to clear.

12.4.38.2 Host Mode

0116 RX1CSR_HOST Rx1 CSR Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AUTOCLEAR	AUTOREQ	DMAREQENAB	PIDERROR	DMAREQMODE	SETREQPKT_TWICE	KEEPERRSTATUS	INCOMPRX	CLRDATATOG	RXSTALL	REQPKT	FLUSHFIFO	DATAERR_NAKTIMEOUT	ERROR	FIFOFULL	RXPKTRDY
Type	RW	RW	RW	RU	RW	A1	RW	A1	A0	A1	RW	A0	A1	A1	RU	A1
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15	AUTOCLEAR	If the CPU sets up this bit then the RxPktRdy bit will be automatically cleared when a packet of RxMaxP bytes has been unloaded from RxFIFO. When packets of smaller than the maximum packet size are unloaded, RxPktRdy will have to be cleared manually. Note: The maximum packet size-3,-2,-1 is just handled like the maximum packet size which is auto cleared by hardware.
14	AUTOREQ	If the CPU sets this bit, the ReqPkt bit will be automatically set when the RxPktRdy bit is cleared. Note: This bit is automatically cleared when a short packet is received.
13	DMAREQENAB	The CPU sets up this bit to enable the DMA request for the Rx endpoint.
12	PIDERROR	ISO transactions only: The core sets up this bit to indicate a PID error in the received packet. Bulk/Interrupt transactions: The setting of this bit is ignored. Write 0 to clear. Note: This register is read -only in the ISO mode but reserved for read/write access in other modes.
11	DMAREQMODE	The CPU sets up this bit to select DMA request mode 1 and clears it to select DMA request mode 0. DMA request mode 1: Rx endpoint interrupt is generated only when DMA request mode 1 and received a short packet. RxDMAReq is generated when receiving a Max-Packet-size packet.DMA request mode 0: No Rx endpoint interrupt. RxDMAReq is generated when RxPktRdy is set.
10	SETREQPKT_TWICE	Indicates the ReqPkt had been set while it is 1 already. Write 0 to clear.
9	KEEPERRSTATUS	Used when the endpoint working with USBQ and in the isochronous mode. When this bit is set, the isochronous error, PIDERROR, INCOMPRX and DATAERROR will be kept and only cleared by SW.
8	INCOMPRX	This bit is set in a Isochronous transfer if the packet in the RxFIFO is incomplete because parts of the data are not received. Cleared when RxPktRdy is cleared or write 0 to clear. Note: In anything other than a high-bandwidth transfer, this bit will always return to 0.
7	CLRDATATOG	The CPU writes 1 to this bit to reset the endpoint data toggle to 0.
6	RXSTALL	When a STALL handshake is received, this bit will be set and an interrupt generated. The CPU clears this bit. Write 0 to clear.

5	REQPKT	The CPU writes 1 to this bit to request an IN transaction. Cleared when RxPktRdy is set.
4	FLUSHFIFO	The CPU writes 1 to this bit to flush the next packet to be read from the endpoint RxFIFO. The RxFIFO pointer is reset and the RxPktRdy bit is cleared. Note: FlushFIFO should only be used when RxPktRdy is set. At other times, it may cause data to be corrupted. Also note that, if the RxFIFO is double buffered, FlushFIFO may need to be set twice to completely clear the RxFIFO.
3	DATAERR_NAKTIMEOUT	In the bulk mode, this bit will be set when the Rx endpoint is halted following the receipt of NAK responses for longer than the time set as the NAK Limit by the RxInterval register. The CPU clears this bit to allow endpoint to continue.Write 0 to clear. When operating in the ISO mode, this bit is set when RxPktRdy is set if the data packet has a CRC or bit-stuff error and cleared when RxPktRdy is cleared.
2	ERROR	The USB sets up this bit when 3 attempts have been made to receive a packet and no data packet has been received.The CPU clears this bit. An interrupt will be generated when the bit is set. Note: This bit is only valid when the Rx endpoint is operating in the bulk or interrupt mode. In the ISO mode, it always returns to 0. Write 0 to clear.
1	FIFOFULL	This bit is set when no more packets can be loaded into the RxFIFO.
0	RXPKTRDY	This bit is set when a data packet has been received (to RxFIFO). The CPU clears this bit when the packet has been unloaded from RxFIFO. An interrupt will be generated when the bit is set. Write 0 to clear.

12.4.39 Rx Count Register

0118 RX1COUNT Rx1 Count Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RXCOUNT													
Type			RU													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
13:0	RXCOUNT	14-bit read-only register that holds the number of received data bytes in the packet in RxFIFO. Note: The value returned changes as FIFO is unloaded and is only valid while RxPktRdy(RxCSR.D0) is set.

12.4.40 TxType Register

11A TX1TYPE Tx1Type Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									TX_SPEED		TX_PROTOCOL		TX_TARGET_EP_NUMBER			
Type									RW		RW		RW			
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:6	TX_SPEED	Operating speed of target device when the core is configured with multipoint option When the core is not configured with the multipoint option, these bits should not be accessed. 2'b00: Unused 2'b01: High 2'b10: Full 2'b11: Low

5:4	TX_PROTOCOL	The CPU sets up this bit to select the required protocol for the Tx endpoint. 2'b00: Illegal 2'b01: Isochronous 2'b10: Bulk 2'b11: Interrupt
3:0	TX_TARGET_EP_NUMBER	The CPU sets up this value to the endpoint number contained in the Tx endpoint descriptor returned to the USB2.0 controller during device enumeration.

12.4.41 TxInterval Register

11B TX1INTERVAL Tx1Interval Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									TX_POLLING_INTERVAL_NAK_LIMIT_M							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	TX_POLLING_INTERVAL_NAK_LIMIT_M	(For host mode only) TxInterval register is an 8-bit register that, for Interrupt and Isochronous transfers, defines the polling interval for the currently-selected Tx endpoint. For Bulk endpoints, this register sets up the number of frames/microframes after which the endpoint should timeout on receiving a stream of NAK responses. There is a TxInterval register for each configured Tx endpoint (except for Endpoint 0). In each case the value that is set defines a number of frames/microframes (high-speed transfers), as follows: Transfer Type | Speed | Valid values (m) | Interpretation Interrupt | Low Speed or Full Speed | 1-255 | Polling interval is m frames. Interrupt | High Speed | 1-16 | Polling interval is 2^(m-1) microframes Isochronous | Full Speed or High Speed | 1-16 | Polling interval is 2^(m-1) frames/microframes Bulk | Full Speed or High Speed | 2-16 | NAK Limit is 2^(m-1) frames/microframes. Note: A value of 0 or 1 disables the NAK timeout function. Note: the register should be set before TxType for Bulk endpoint.

12.4.42 RxType Register

11C RX1TYPE Rx1Type Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXSPEED		RX_PROTOCOL		RX_TARGET_EP_NUMBER			
Type									RW		RW		RW			
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:6	RXSPEED	Operating speed of target device when the core is configured with multipoint option When the core is not configured with the multipoint option, these bits should not be accessed. 2'b00: Unused 2'b01: High 2'b10: Full 2'b11: Low
5:4	RX_PROTOCOL	The CPU sets up this bit to select the required protocol for the Tx endpoint. 2'b00: Illegal 2'b01: Isochronous 2'b10: Bulk 2'b11: Interrupt
3:0	RX_TARGET_EP_NUMBER	The CPU sets up this value to the endpoint number contained in the Tx endpoint descriptor returned to the USB2.0 controller during device enumeration.

12.4.43 RxInterval Register

11D RX1INTERVAL Rx1Interval Register 00

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RX_POLLING_INTERVAL_NAK_LIMIT_M							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	RX_POLLING_INTERVAL_NAK_LIMIT_M	RxInterval register is an 8-bit register that, for Interrupt and Isochronous transfers, defines the polling interval for the currently-selected Rx endpoint. For Bulk endpoints, this register sets up the number of frames/microframes after which the endpoint should timeout on receiving a stream of NAK responses. There is a RxInterval register for each configured Rx endpoint (except for Endpoint 0). RX POLLING INTERVAL/NAK LIMIT (M), (host mode only) In each case the value that is set defines a number of frames/microframes (high-speed transfers), as follows: Transfer Type Speed Valid values (m) Interpretation Interrupt Low Speed or Full Speed 1 - 255 Polling interval is m frames. High Speed 1 - 16 Polling interval is 2(m-1) microframes Isochronous Full Speed or High Speed 1 - 16 Polling interval is 2(m-1) frames/microframes Bulk Full Speed or High Speed 2 - 16 NAK Limit is 2(m-1) frames/microframes. Note: Value of 0 or 1 disables the NAK timeout function. Note: the register should be set before RxType for Bulk endpoint.

12.4.44 Configured FIFO Size Register

12.4.44.1 Peripheral Mode

11F FIFOSIZE1 EP1 Configured FIFO Size Register NA

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXFIFOSIZE				TXFIFOSIZE			
Type									DC				DC			
Reset									X	X	X	X	X	X	X	X

Bit(s)	Name	Description
7:4	RXFIFOSIZE	Indicates the RxFIFO size of 2^n bytes, e.g. value 10 means 2^10 = 1,024 bytes
3:0	TXFIFOSIZE	Indicates the TxFIFO size of 2^n bytes, e.g.: value 10 means 2^10 = 1,024 bytes.

The configured FIFO size register value is undefined if Dynamic FIFO is configured. The existing endpoints are default to HW configured value 10. The non-exist endpoints are default 0.
If the dynamic FIFO is not configured, the register represents the corresponding endpoint's FIFO size setting.

USB+0120h ~ USB+012Fh: Endpoint 2 registers and their behaviors are the same as Endpoint 1.
USB+0130h ~ USB+013Fh: Endpoint 3 registers and their behaviors are the same as Endpoint 1.
USB+0140h ~ USB+014Fh: Endpoint 4 registers and their behaviors are the same as Endpoint 1.
USB+0150h ~ USB+015Fh: Endpoint 5 registers and their behaviors are the same as Endpoint 1.
USB+0160h ~ USB+016Fh: Endpoint 6 registers and their behaviors are the same as Endpoint 1.

USB+0170h ~ USB+017Fh: Endpoint 7 registers and their behaviors are the same as Endpoint 1.
USB+0180h ~ USB+018Fh: Endpoint 8 registers and their behaviors are the same as Endpoint 1.

12.4.45 DMA Interrupt Status Register (Byte Access)

00000200 DMA_INTR DMA Interrupt Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMA_INTR_UNMASK_SET								DMA_INTR_UNMASK_CLEAR							
Type	W1								W1							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMA_INTR_UNMASK								DMA_INTR_STATUS							
Type	RU								W1C							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DMA_INTR_UNMASK_SET	Sets DMA_INTR_UNMASK to 1
23:16	DMA_INTR_UNMASK_CLEAR	Clears DMA_INTR_UNMASK to 0
15:8	DMA_INTR_UNMASK	Unmasks DMA interrupts The DMA interrupt will be generated when both DMA_INTR_UNMASK and DMA_INTR_STATUS are 1.
7:0	DMA_INTR_STATUS	Indicates DMA complete interrupt status, one bit per DMA channel implemented Bit 0 is used for DMA channel 1; bit 1 is used for DMA channel 2 etc. Write 1 to clear it. Note: DMA interrupt will be asserted after disabling the DMA enable when receiving a null packet even thought DMA_COUNT_M still does not achieve 0.

12.4.46 DMA Limiter Register (Word Access)

00000210 DMA_LIMITER DMA Limiter Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DMA_LIMITER							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	DMA_LIMITER	This register suppresses the bus utilization of the DMA channel. The value is from 0 to 255. 0 means no limitation, and 255 means totally banned. The value between 0 and 255 means certain DMA is permitted to use AHB every (4 x n) AHB clock cycles. Note: It is not recommended to limit bus utilization of the DMA channels because this will increase the latency of response to the masters, and the transfer rate decreases as well. Before using it, the programmer must make sure that the bus masters have some protective mechanism to avoid entering wrong states.

12.4.47 DMA Configuration Register (Word Access)

00000220 DMA_CONFIG DMA Configuration Register 00000004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DMA_ACTIVE_EN		AHB_HPROT_2_EN			DMAQ_CHAN_SEL			MCU_LOCK_SEL		AHBWAIT_SEL	BOUNDARY_1K_CROSS_EN
Type					RW		RW			RW			RW		RW	RW
Reset					0	0	0	0		0	0	0	0	1	0	0

Bit(s)	Name	Description
11:10	DMA_ACTIVE_EN	The two bits control usb_active. 2'b00: usb_active depends on all DMAEN of DMA channel control register . 2'b01: usb_active ties to 1 2'b10: usb_active ties to 0 2'b11: usb_active depends on ep_active, dma_active and all DMAEN of DMA channel control register(OR logic)
9:8	AHB_HPROT_2_EN	The two bits control the AHB master interface HPROT2 function operateing in non-bufferable/bufferable/last transfer non-bufferable mode. 2'b00: All write transfers of a burst will be accessed by bufferable mode except for the last transfer of a burst . 2'b01: AHB master HPROT2 is always accessed by non-bufferable mode. 2'b10: AHB master HPROT2 is always accessed by bufferable mode 2'b11: Reserved
6:4	DMAQ_CHAN_SEL	Selects DMA channel used by USB_DMAQ if it is available It will not affect if USB_DMAQ is not available.
3:2	MCU_LOCK_SEL	Issues fix TXQ and FIFO PIO operation concurrent (HW patch & Bug Fix only. Not a normal function.) 2'b00: No lock 2'b01: Lock 1 level (IP clock) 2'b10: lock 2 level (IP clock) 2'b11: No lock
1	AHBWAIT_SEL	Selects AHBWAIT behavior Set to 1 to return to old DMA master AHB wait condition. This bit is used to test DMA FIFO overflow bug.
0	BOUNDARY_1K_CROSS_EN	Enables 1k boundary page crossing Set to 1 to force burst transfer regardless 1k boundary crossing. Note: It will violate AHB 1k boundary specification but gain some bus performance.

12.4.48 DMA Channel M Control Register (Word Access)

0204 DMA_CNTL_0 DMA Channel 0 Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DMAABORT	DMAREQUEST	DMACHEN	BURST_MODE		BUSERR	ENDPNT				INTEN	DMAMODE	DMADIR	DMAEN
Type			A0	RU	RU	RW		RU	RW				RW	RW	RW	Other
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
13	DMAABORT	If SW needs abort the current DMA transfer, set DMAABORT=1 & DMAEN=0, after the transfer is aborted completely, DMA interrupt occurs.
12	DMAREQUEST	Indicate whether DMA transfer is requesting or not. 1 means DMA is requesting.

11	DMACHEN	DMA channel enable monitor bit
10:9	BURST_MODE	2'b00: Burst Mode 0 : Bursts of unspecified length 2'b01: Burst Mode 1 : INCR4 or unspecified length 2'b10: Burst Mode 2 : INCR8, INCR4 or unspecified length 2'b11: Burst Mode 3 : INCR16, INCR8, INCR4 or unspecified length
8	BUSERR	Bus error
7:4	ENDPNT	Endpoint DMA will transfer with
3	INTEN	Enables interrupt
2	DMAMODE	DMA mode DMA mode 0: Single packet operation DMA mode 1: Multi packets operation, with the configuraiton of DMAReqMode in RXCSR bit 11. DMA mode 1 supports both known size or unknown size transaction.
1	DMADIR	Direction 0 : DMA write (Rx endpoint) 1 : DMA read (Tx endpoint)
0	DMAEN	Enables DMA The bit will be cleared when the DMA transfer is completed. Programmers should not disable DMA_EN before the transfer is completed. If the programeers disables DMA_EN when transferring, DMA will not stop immediately untill the last bus transfer is completed.

The DMA Channel count information can be read from RAMINFO. The legal values of M are 1 ~ 8. DMA_CNTL_M offset = DMA_CNTL_0 offset + (10h*M-1).

12.4.49 DMA Channel M Address Register (Word Access)

00000208 DMA_ADDR_0 DMA Channel 0 Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMA_ADDR_0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMA_ADDR_0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DMA_ADDR_0	32-bit DMA start address Updated (increased) by USB2.0 controller automatically while multiple packet DMA (DMA mode = 1) is used.

The DMA Channel count information can be read from RAMINFO. The legal values of M are 1 ~ 8. DMA_ADDR_M offset = DMA_ ADDR_0 offset + (10h*M-1).

12.4.50 DMA Channel M Byte Count Register (Word Access)

0000020C DMA_COUNT_0 DMA Channel 0 Byte Count Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									DMA_COUNT_0[23:16]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMA_COUNT_0[15:0]															

Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23:0	DMA_COUNT_0	24-bit DMA transfer count with byte unit Updated (decreased) by USB2.0 controller automatically while each packet is transferred.

The DMA Channel count information can be read from RAMINFO. The legal values of M are 1 ~ 8. DMA_COUNT_M offset = DMA_ COUNT_0 offset + (10h*M-1).

12.4.51 EPn RxPktCount Register (Word Access)

0300 EP0RXPKTCOUNT EP0 RxPktCount Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EP0RXPKTCOUNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	EP0RXPKTCOUNT	Sets up the number of packets of Rx Endpoint n size MaxP to be transferred in a block transfer Only used in Host mode when AutoReq is set. Has no effect in Peripheral mode or when AutoReq is not set. RqPktCount (host mode only) For each Rx Endpoint 1 - 15, the USB2.0 controller provides a 16-bit RqPktCount register. This read/write register is used in the host mode to specify the number of packets that are to be transferred in a block transfer of one or more Bulk packets of length MaxP to Rx Endpoint n. The core uses the value recorded in this register to etermine the number of requests to issue where the AutoReq option (included in the RxCSR register) has been set.Note: Multiple packets combined into a single bulk packet within the FIFO count as one packet.

The RX Endpoint count information can be read from EPINFO. The legal values of N are 0 ~ 15. EP(N)RXPKTCOUNT offset = EP0RXPKTCOUNT offset + (4h*N).

12.4.52 Transmit Endpoint n Function Address (Word Access)

0480 T0FUNCADDR T0 Function Address Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										T0FuncAddr						
Type										RW						
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6:0	T0FuncAddr	7 -bit read/write register recording the address of the target function to be accessed through the associated endpoint (Endpoint 0) Needs to be defined for each Tx endpoint that is used. Note: TxFuncAddr must be defined for Endpoint 0.

The TX Endpoint count information can be read from EPINFO. The legal values of N are 0 ~ 15.

T(N)FUNCADDR offset = T0FUNCADDR offset + (8h*N).

12.4.53 Transmit Endpoint n Hub/Port Address (Word Access)

0482 T0HUBADDR T0 HUB Address Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		T0PortAddr							T0MultiTranslator	T0HUBAddr						
Type		RW							RW	RW						
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
14:8	T0PortAddr	Only needs to be written where a full or low speed device is connected to Tx Endpoint n via a high speed USB 2.0 hub carrying out the necessary transaction translation In such circumstances, the 7-bit read/write registers is used to record the port of that USB 2.0 hub through which the target associated with the endpoint is accessed. Note: If Endpoint 0 is connected to a hub, TxHubPort must be defined. Only valid if RTL defines USB_HUB. This register is only valid @ FS/LS is plugged in at HUB.
7	T0MultiTranslator	Bit 8 should record whether the hub has multiple transaction translator . Only valid if RTL defines USB_HUB. This register is only valid @ FS/LS is plugged in at HUB . 1'b0: Single transaction translator 1'b1: Multiple transaction translators
6:0	T0HUBAddr	8 -bit read/write register to be written where a full or low speed device is connected to Tx Endpoint n via a high -speed USB2.0 hub carrying out the necessary transaction translation to convert between high speed transmission and full/low speed transmission The lower 7 bits record the address of this USB 2.0 HUB. Note: If Endpoint 0 is connected to a hub, TxHubAddr must be defined for EP0. Only valid if RTL defines USB_HUB . This register is only valid @ FS/LS is plugged in at HUB .

The TX Endpoint count information can be read from EPINFO. The legal values of N are 0 ~ 15.
T(N)HUBADDR offset = T0HUBADDR offset + (8h*N).

12.4.54 Receive Endpoint n Function Address (Word Access)

0484 R0FUNCADDR R0 Function Address Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										R0FuncAddr						
Type										RW						
Reset										0	0	0	0	0	0	0

Bit(s)	Name	Description
6:0	R0FuncAddr	7 -bit read/write register recording the address of the target function to be accessed through the associated endpoint (Endpoint 0) Needs to be defined for each Rx endpoint that is used. Note: RxFuncAddr does not exist on EP0.

The RX Endpoint count information can be read from EPINFO. The legal values of N are 0 ~ 15.
R(N)FUNCADDR offset = R0FUNCADDR offset + (8h*N).

12.4.55 Receive Endpoint n Hub/Port Address (Word Access)

0486 R0HUBADDR R0 HUB Address Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		R0PortAddr							R0MultiTranslator	R0HUBAddr						
Type		RW							RW	RW						
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
14:8	R0PortAddr	Only needs to be written where a full or low speed device is connected to Rx Endpoint n via a high -speed USB 2.0 hub carrying out the necessary transaction translation In such circumstances, the 7-bit read/write register is used to record the port of that USB 2.0 hub through which the target associated with the endpoint is accessed. Note: The RxPortAddr doest not exist on EP0. Only valid if RTL defines USB_HUB. This register is only valid @ FS/LS is plugged in at HUB.
7	R0MultiTranslator	Bit 8 records whether the hub has multiple transaction translator Only valid if RTL defines USB_HUB. This register is only valid @ FS/LS is plugged in at HUB . 1'b0: Single transaction translator 1'b1: Multiple transaction translators
6:0	R0HUBAddr	8 -bit read/write register needing to be written where a full or low speed device is connected to Rx Endpoint n via a high -speed USB2.0 hub carrying out the necessary transaction translation to convert between high -speed transmission and full/low -speed transmission The lower 7 bits record the address of this USB 2.0 HUB. Note: RxHubAddr does not exist on EP0. Only valid if RTL defines USB_HUB . This register is only valid @ FS/LS is plugged in at HUB.

The RX Endpoint count information can be read from EPINFO. The legal values of N are 0 ~ 15. R(N)HUBADDR offset = R0HUBADDR offset + (8h*N).

12.5 USBQMU Register Definition

USBQMU Base Address

USB P0: +11200000h

USB P1: +11270000h

12.5.1 Queue Control Registers (QCR0-QCR3)

00000800 QCR0 Queue Control Register 0 80000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Q_CS16B_EN								T7Q_CS_EN	T6Q_CS_EN	T5Q_CS_EN	T4Q_CS_EN	T3Q_CS_EN	T2Q_CS_EN	T1Q_CS_EN	T0Q_CS_EN
Type	RW								RW	RW	RW	RW	RW	RW	RW	RW

Reset	1								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									R7Q_CS_EN	R6Q_CS_EN	R5Q_CS_EN	R4Q_CS_EN	R3Q_CS_EN	R2Q_CS_EN	R1Q_CS_EN	R0Q_CS_EN
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	Q_CS16B_EN	Tx 0 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable 16 bytes checksum. (wimax only) 1'b0: Disable 16 bytes checksum, therefore, 12 bytes checksum for GPD and BD. 1'b1: Enable 16 bytes checksum, therefore, 16 bytes checksum for GPD and BD.
23	T7Q_CS_EN	Transmit Queue (T7Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
22	T6Q_CS_EN	Transmit Queue (T6Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
21	T5Q_CS_EN	Transmit Queue (T5Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
20	T4Q_CS_EN	Transmit Queue (T4Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
19	T3Q_CS_EN	Transmit Queue (T3Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
18	T2Q_CS_EN	Transmit Queue (T2Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
17	T1Q_CS_EN	Transmit Queue (T1Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
16	T0Q_CS_EN	Transmit Queue (T0Q) Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for TXQ. 1'b0: Disable the descriptor checksum validation function for TxQ. 1'b1: Enable the descriptor checksum validation function for TxQ.
7	R7Q_CS_EN	RxQ Receive 7 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
6	R6Q_CS_EN	RxQ Receive 6 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
5	R5Q_CS_EN	RxQ Receive 5 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.

4	R4Q_CS_EN	RxQ Receive 4 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
3	R3Q_CS_EN	RxQ Receive 3 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
2	R2Q_CS_EN	RxQ Receive 2 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
1	R1Q_CS_EN	RxQ Receive 1 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.
0	R0Q_CS_EN	RxQ Receive 0 Queue Descriptor Checksum Validation Enable This bit is used to enable or disable the descriptor checksum validation function for RXQ. 1'b0: Disable the descriptor checksum validation function for RXQ. 1'b1: Enable the descriptor checksum validation function for RXQ.

00000804 QCR1 Queue Control Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		R0Q_DATA_BUF_SIZE				R6Q_DATA_BUF_SIZE				R2Q_DATA_BUF_SIZE				R5Q_DATA_BUF_SIZE		
Type		RW				RW				RW				RW		
Reset		0	0	0		0	0	0		0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		R1Q_DATA_BUF_SIZE				R4Q_DATA_BUF_SIZE				R7Q_DATA_BUF_SIZE				R3Q_DATA_BUF_SIZE		
Type		RW				RW				RW				RW		
Reset		0	0	0		0	0	0		0	0	0		0	0	0

Bit(s)	Name	Description
30:28	R0Q_DATA_BUF_SIZE	R0Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
26:24	R6Q_DATA_BUF_SIZE	R6Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes

22:20	R2Q_DATA_BUF_SIZE	R2Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
18:16	R5Q_DATA_BUF_SIZE	R5Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
14:12	R1Q_DATA_BUF_SIZE	R1Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
10:8	R4Q_DATA_BUF_SIZE	R4Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
6:4	R7Q_DATA_BUF_SIZE	R7Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes
2:0	R3Q_DATA_BUF_SIZE	R3Q Queue Data Buffer Size This field indicates the size of the data buffers used in xxx queue. Each bit in this field represents a specific data buffer size. The field definition is shown as follows. 3'b000: 32 bytes 3'b001: 64 bytes 3'b010: 128 bytes 3'b011: 256 bytes 3'b100: 512 bytes 3'b101: 1K bytes 3'b110: 2K bytes 3'b111: 4K bytes

00000808 QCR2 Queue Control Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CFG_TX_ZLP7	CFG_TX_ZLP6	CFG_TX_ZLP5	CFG_TX_ZLP4	CFG_TX_ZLP3	CFG_TX_ZLP2	CFG_TX_ZLP1	CFG_TX_ZLP0	CFG_TX_MULTIEP7	CFG_TX_MULTIEP6	CFG_TX_MULTIEP5	CFG_TX_MULTIEP4	CFG_TX_MULTIEP3	CFG_TX_MULTIEP2	CFG_TX_MULTIEP1	CFG_TX_MULTIEP0
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															CFG_TX_PADDING	CFG_TX_SDUHDR
Type															RW	RW
Reset															0	0

Bit(s)	Name	Description
31	CFG_TX_ZLP7	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
30	CFG_TX_ZLP6	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
29	CFG_TX_ZLP5	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
28	CFG_TX_ZLP4	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
27	CFG_TX_ZLP3	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
26	CFG_TX_ZLP2	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
25	CFG_TX_ZLP1	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
24	CFG_TX_ZLP0	Transmit ZLP configuration When cfg_tx_multiep = 1, sending of ZLP is based on ZLP field in GPD. When cfg_tx_multiep = 0, this bit determines whether to send ZLP or not. 1'b0: Do not send ZLP packet 1'b1: Send ZLP packet
23	CFG_TX_MULTIEP7	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not.

		When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
22	CFG_TX_MULTIEP6	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
21	CFG_TX_MULTIEP5	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
20	CFG_TX_MULTIEP4	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
19	CFG_TX_MULTIEP3	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
18	CFG_TX_MULTIEP2	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
17	CFG_TX_MULTIEP1	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
16	CFG_TX_MULTIEP0	Multi endpoint configuration This bit determines whether the endpoint is connected to multiple endpoint or not. When this bit is set, the HW will also parse TGL and Target Endpoint Number to determine whether or not to set data toggle and what endpoint number to program this GPD for. (host mode only) 1'b0: Single endpoint 1'b1: Multi-endpoint
1	CFG_TX_PADDING	Transmit Padding Configuration This bit determines sending of padding bytes to make each GPD word aligned. (cfg_sduhdr=1 only. Reserved when cfg_sduhdr=0) 1'b0: Do not send padding 1'b1: Send padding
0	CFG_TX_SDUHDR	Transmit SDU header configuration This bit determines sending of 4 byte pre-defined SDU header. (wimax only) 1'b0: Do not send SDU header 1'b1: Send SDU header

0000080C QCR3 Queue Control Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CFG_RX_ZLP7	CFG_RX_ZLP6	CFG_RX_ZLP5	CFG_RX_ZLP4	CFG_RX_ZLP3	CFG_RX_ZLP2	CFG_RX_ZLP1	CFG_RX_ZLP0	CFG_RX_MULTIEP7	CFG_RX_MULTIEP6	CFG_RX_MULTIEP5	CFG_RX_MULTIEP4	CFG_RX_MULTIEP3	CFG_RX_MULTIEP2	CFG_RX_MULTIEP1	CFG_RX_MULTIEP0
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFG_RX_COZ7	CFG_RX_COZ6	CFG_RX_COZ5	CFG_RX_COZ4	CFG_RX_COZ3	CFG_RX_COZ2	CFG_RX_COZ1	CFG_RX_COZ0							CFG_RX_MAXPHB	CFG_RX_SDUHDR
Type	RW	RW	RW	RW	RW	RW	RW	RW							RW	RW
Reset	0	0	0	0	0	0	0	0							0	0

Bit(s)	Name	Description
31	CFG_RX_ZLP7	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
30	CFG_RX_ZLP6	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
29	CFG_RX_ZLP5	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
28	CFG_RX_ZLP4	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
27	CFG_RX_ZLP3	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
26	CFG_RX_ZLP2	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
25	CFG_RX_ZLP1	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
24	CFG_RX_ZLP0	Receive ZLP configuration This bit determines whether or not to expect ZLP if packets end on max packet boundary (only used when cfg_rx_multiep = 0) 1'b0: Do not expect ZLP 1'b1: Expect ZLP
23	CFG_RX_MULTIEP7	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint

		1'b1: Multiple endpoint
22	CFG_RX_MULTIEP6	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
21	CFG_RX_MULTIEP5	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
20	CFG_RX_MULTIEP4	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
19	CFG_RX_MULTIEP3	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
18	CFG_RX_MULTIEP2	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
17	CFG_RX_MULTIEP1	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
16	CFG_RX_MULTIEP0	Receive multiple endpoint configuration This bit determines whether or not to parse the GPD for data toggle and endpoint number on receive side (Host mode only) 1'b0: Single endpoint 1'b1: Multiple endpoint
15	CFG_RX_COZ7	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
14	CFG_RX_COZ6	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
13	CFG_RX_COZ5	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
12	CFG_RX_COZ4	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
11	CFG_RX_COZ3	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
10	CFG_RX_COZ2	Empty data receive ZLP and jump to next GDP configuration

		This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
9	CFG_RX_COZ1	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
8	CFG_RX_COZ0	Empty data receive ZLP and jump to next GDP configuration This bit determines whether or not jump to next GPD when ZLP is received and current GPD still have no data. 1'b0: Do not jump to next GPD 1'b1: Jump to next GPD
1	CFG_RX_MAXPHB	This bit is only used in high bind-width endpoints. Please refer to USB MAC register RxMaxP for the meaning of M. For example, if M=2, and only DATA1&DATA0 (with maxpayload for each packet), it is treated as short packet if CFG_RX_MAXPHB=0, and treated as maxpkt if CFG_RX_MAXPHB=1. 1'b0: only if (M+1) packets with size of maxpayload are received, it is treated as maxpkt for high bind-width endpoints 1'b1: even less than (M+1) packets are received, but each packet size is maxpayload, it is still treated as maxpkt.
0	CFG_RX_SDUHDR	Receive SDU header configuration This bit determines whether or not to parse an SDU header on receive side 1'b0: No SDU header 1'b.1: SDU header

12.5.2 RX Queue Command and Status Registers (RQCSRn)

The RX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

RQCSRn = RQCSR0 offset + 0x10h*(n-1)

00000810 RQCSR0 RX Queue Command and Status Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXQ_ACTIVE													RXQ_STOP	RXQ_RESUME	RXQ_START
Type	RU													A0	A0	A0
Reset	0													0	0	0

Bit(s)	Name	Description
15	RXQ_ACTIVE	RX Queue 0 Active This bit is used to indicate whether the RX Queue is in the active state. When the HIF Controller is reset, the queue is in the inactive state by default. After receiving a START or RESUME command and executing it without error, the queue enters the active state. When the queue is empty or stopped by a STOP command, it returns to the inactive state. 1'b0: Inactive 1'b1: Active
2	RXQ_STOP	RX Queue 0 Stop This bit is used to issue a STOP command to the RX Queue.

		In the case of RBEQ_STOP, all 4 best effort queues are stopped together.
1	RXQ_RESUME	RX Queue 0 Resume This bit is used to issue a RESUME command to the RX Queue.
0	RXQ_START	RX Queue 1 Start This bit is used to issue a START command to the RX Queue.

12.5.3 RX Queue Starting Address Registers (RQSARn)

The RX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

RQSARn = RQSAR0 offset + 0x10h*(n-1)

00000814 RQSAR0 RX Queue Starting Address Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RXQ_START_ADDR0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXQ_START_ADDR0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RXQ_START_ADDR0	

12.5.4 RX Queue Current Pointer Registers (RQCPRn)

The RX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

RQCPRn = RQCPR0 offset + 0x10h*(n-1)

00000818 RQCPR0 RX Queue Current Pointer Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RXQ_CURR_PTR0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXQ_CURR_PTR0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RXQ_CURR_PTR0	

12.5.5 RX Timeout Register n (RQTRn)

The RX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

RQTRn = RQTR0 offset + 0x10h*(n-1)

00000890 RQTR0 RX Timeout Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXQ_TIMEOUT0							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	RXQ_TIMEOUT0	RX Timeout Register This 8 bit register sets the number of 8K cycles for USB Queue to wait after receiving USB packet before it treats the GPD as done and writes length and HWO back into GPD.

12.5.6 RX Queue Last Done Pointer Register n (RQLDPRn)

The RX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

RQLDPRn = RQLDPR0 offset + 0x10h*(n-1)

00000900 RQLDPR0 RX Queue Last Done Pointer Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RXQ_LAST_DONE_PTR0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXQ_LAST_DONE_PTR0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RXQ_LAST_DONE_PTR0	Last Done Pointer Register. As the QMU complete the last transaction in RXQ queue, it will update this register to the last-done pointer as the same time as updating RQCPRn, and the FW can see this pointer to fast de-queue after sense the RQCPRn been changed. This pointer will be reset to 0 as initialization, and it will be forced to the starting address (RQSARn) after the "Start" command been issued.

12.5.7 Tx n Queue Command and Status Register (TQCSRn)

The TX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

TQCSRn = TQCSR0 offset + 0x10h*(n-1)

00000A00 TQCSR0 TX Queue Command and Status Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXQ_ACTIVE													TXQ_STOP	TXQ_RESUME	TXQ_START
Type	RU													A0	A0	A0
Reset	0													0	0	0

Bit(s)	Name	Description
15	TXQ_ACTIVE	TX Queue 0 Active This bit is used to indicate the current state of the TQ. 1'b0: Inactive 1'b1: Active
2	TXQ_STOP	TX Queue 0 Stop This bit is used to stop the operation of the TQ. Note that in HIF-SDIO, this Queue STOP command clears TX FIFO and clear data count in TQDCR.
1	TXQ_RESUME	TX Queue 0 Resume This bit is used to resume the operation of the TQ.
0	TXQ_START	TX Queue 0 Start This bit is used to start the operation of the TQ.

The TX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

TQCSRn = TQCSR0 offset + 0x10h*(n-1)

12.5.8 TX n Queue Starting Address Register (TQSARn)

The TX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

TQSARn = TQSAR0 offset + 0x10h*(n-1)

00000A04 TQSAR0 TX Queue Starting Address Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TXQ_START_ADDR0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXQ_START_ADDR0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TXQ_START_ADDR0	TX Queue Starting Address The starting address of the T0Q.

12.5.9 TX n Queue Current Pointer Register (TQCPRn)

The TX Queue count information can be read from QMU_HWVER register. The legal values of n is 1 ~ 8.

TQCPRn = TQCPR0 offset + 0x10h*(n-1)

00000A08 TQCPR0 TX Queue Current Pointer Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TXQ_CURR_PTR0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TXQ_CURR_PTR0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	Name	Description
:	TXQ_CURR_PTR0	TX Queue Current Pointer The current pointer of the T0Q.

12.5.10 USB General Control and Status Register (USBGCSR)

00000B00 USBGCSR USB General Control and Status Register 00008000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BUS_RST_QUE							QUE_EN_HIF_CMD	QUE_EN_RX7	QUE_EN_RX6	QUE_EN_RX5	QUE_EN_RX4	QUE_EN_RX3	QUE_EN_RX2	QUE_EN_RX1	QUE_EN_RX0
Type	RW							RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0							0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QUE_CG_DIS							QUE_EN_HIF_EVT	QUE_EN_TX7	QUE_EN_TX6	QUE_EN_TX5	QUE_EN_TX4	QUE_EN_TX3	QUE_EN_TX2	QUE_EN_TX1	QUE_EN_TX0
Type	RW							RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1							0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31	BUS_RST_QUE	Bus Resets Queue When this bit is set to 1, a bus reset on USB will also trigger a card reset. Wimax only. Used for debugging 1'b0: Card does not reset on bus reset 1'b1: Card resets on bus reset
24	QUE_EN_HIF_CMD	
23	QUE_EN_RX7	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
22	QUE_EN_RX6	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
21	QUE_EN_RX5	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
20	QUE_EN_RX4	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
19	QUE_EN_RX3	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
18	QUE_EN_RX2	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
17	QUE_EN_RX1	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
16	QUE_EN_RX0	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
15	QUE_CG_DIS	The EPQ control can be turned off if it is not serviced to enhance dynamic power. 1'b0: The CG cells inserted in QMU is enabled, so the clock can be turned off dynamicly.

		1'b1: The CG cells in QMU is disabled.
8	QUE_EN_HIF_EVT	
7	QUE_EN_TX7	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
6	QUE_EN_TX6	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
5	QUE_EN_TX5	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
4	QUE_EN_TX4	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
3	QUE_EN_TX3	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
2	QUE_EN_TX2	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
1	QUE_EN_TX1	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU
0	QUE_EN_TX0	USB QMU Enable This field is used to reflect the USB QMU control USB DMA or not. Each bit corresponds to each endpoint in the USB. If all QUE_EN_XX bits are set to zero, then the USB DMA channel is allowed to be directly used for software. 1'b0: USB Queue is Controlled by Software 1'b1: USB Queue is Controlled by QMU

12.5.11 USB Firmware Register 1 (USB_FW1)

00000B04 USB_FW1 USB Firmware Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FW1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FW1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FW1	

12.5.12 USB Firmware Register 2 (USB_FW2)

00000B08 USB_FW2 USB Firmware Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FW2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FW2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FW2	

12.5.13 TnQ USB Stream Constraint Register (T0QUSBSC)

00000B80 T0QUSBSC T0Q USB Stream Constraints 00000020

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									SPCOUNT							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						SSIZE										
Type						RW										
Reset						0	0	0	0	0	1	0	0	0	0	0

Bit(s)	Name	Description
23:16	SPCOUNT	USB QMU Maximum Stream PDU Count

		0x00 TXQ is packet mode Example: 0x10 Maximum stream PDU count = 0x10 = 16 If there are already 16 GPD send to host, QMU will stop this stream regardless of the constraint of empty or stream size.
10:0	SSIZE	USB QMU Maximum Stream Size This register is not valid when SPCOUNT is smaller than 2. The unit of this register is 64 bytes. Example: 0x40 Maximum stream size = (0x40)*(64) = 4096 bytes

12.5.14 QMU Interrupt Status and Acknowledgement Register (QISAR)

00000C00 QISAR QMU Interrupt Status and Acknowledgement Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				TxEP_ERR_INT	TxQ_ERR_INT	RxEP_ERR_INT	RxQ_ERR_INT	RxBEQ_ERR_INT	RBEQ_DEQ_INT	RBEQ_EMPTYBEQ3_INT	RBEQ_EMPTYBEQ2_INT	RBEQ_EMPTYBEQ1_INT	RBEQ_EMPTYBEQ0_INT	RBEQ_DONE_INT	RxQ_EMPTY_INT	TxQ_EMPTY_INT
Type				W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_DONE_INT	R6Q_DONE_INT	R5Q_DONE_INT	R4Q_DONE_INT	R3Q_DONE_INT	R2Q_DONE_INT	R1Q_DONE_INT	R0Q_DONE_INT	T7Q_DONE_INT	T6Q_DONE_INT	T5Q_DONE_INT	T4Q_DONE_INT	T3Q_DONE_INT	T2Q_DONE_INT	T1Q_DONE_INT	T0Q_DONE_INT
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	TxEP_ERR_INT	TxEP Error Interrupt This bit is set to 1 when a USB packet error occurs in the USB MAC and signifies to SW that the HW Queue may need to be stopped and restarted. The corresponding endpoint for the error occurred is specified in the Tx Endpoint Error Indication Register.
27	TxQ_ERR_INT	TxQ Error Interrupt This bit is set to 1 when the TxQ detects errors and stops its operation. The corresponding reasons for the errors are specified in the Tx Queue Error Indication Register.
26	RxEP_ERR_INT	RxEP Error Interrupt This bit is set to 1 when a USB packet error occurs in the USB MAC and signifies to SW that the HW Queue may need to be stopped and restarted. The corresponding endpoint for the error occurred is specified in the Rx Endpoint Error Indication Register.
25	RxQ_ERR_INT	RxQ Error Interrupt This bit is set to 1 when the RxQ detects errors and stops its operation. The corresponding reasons for the errors are specified in the Command Queue Error Indication Register.
24	RxBEQ_ERR_INT	
23	RBEQ_DEQ_INT	
22	RBEQ_EMPTYBEQ3_INT	
21	RBEQ_EMPTYBEQ2_INT	
20	RBEQ_EMPTYBEQ1_INT	
19	RBEQ_EMPTYBEQ0_INT	

18	RBEQ_DONE_INT	
17	RxQ_EMPTY_INT	Rx Queue (RxQ) Empty Interrupt This bit is set to 1 when a RxQ is empty.
16	TxQ_EMPTY_INT	Tx Queue Empty Interrupt This bit is set to 1 when a TxQ is empty.
15	R7Q_DONE_INT	Rx Queue (R7Q) Done Interrupt This bit is set to 1 when the R7Q has acquired one unit of command data from the host.
14	R6Q_DONE_INT	Rx Queue (R6Q) Done Interrupt This bit is set to 1 when the R6Q has acquired one unit of command data from the host.
13	R5Q_DONE_INT	Rx Queue (R5Q) Done Interrupt This bit is set to 1 when the R5Q has acquired one unit of command data from the host.
12	R4Q_DONE_INT	Rx Queue (R4Q) Done Interrupt This bit is set to 1 when the R4Q has acquired one unit of command data from the host.
11	R3Q_DONE_INT	Rx Queue (R3Q) Done Interrupt This bit is set to 1 when the R3Q has acquired one unit of command data from the host.
10	R2Q_DONE_INT	Rx Queue (R2Q) Done Interrupt This bit is set to 1 when the R2Q has acquired one unit of command data from the host.
9	R1Q_DONE_INT	Rx Queue (R1Q) Done Interrupt This bit is set to 1 when the R1Q has acquired one unit of command data from the host.
8	R0Q_DONE_INT	Rx Queue (R0Q) Done Interrupt This bit is set to 1 when the R0Q has acquired one unit of command data from the host.
7	T7Q_DONE_INT	Tx Queue (T7Q) Done Interrupt This bit is set to 1 when the T7Q has passed one unit of Tx data to the host.
6	T6Q_DONE_INT	Tx Queue (T6Q) Done Interrupt This bit is set to 1 when the T6Q has passed one unit of Tx data to the host.
5	T5Q_DONE_INT	Tx Queue (T5Q) Done Interrupt This bit is set to 1 when the T5Q has passed one unit of Tx data to the host.
4	T4Q_DONE_INT	Tx Queue (T4Q) Done Interrupt This bit is set to 1 when the T4Q has passed one unit of Tx data to the host.
3	T3Q_DONE_INT	Tx Queue (T3Q) Done Interrupt This bit is set to 1 when the T3Q has passed one unit of Tx data to the host.
2	T2Q_DONE_INT	Tx Queue (T2Q) Done Interrupt This bit is set to 1 when the T2Q has passed one unit of Tx data to the host.
1	T1Q_DONE_INT	Tx Queue (T1Q) Done Interrupt This bit is set to 1 when the T1Q has passed one unit of Tx data to the host.
0	T0Q_DONE_INT	Tx Queue (T0Q) Done Interrupt This bit is set to 1 when the T0Q has passed one unit of Tx data to the host.

This register is used both by the Host Interface QMU to indicate which types of interrupts have occurred and by the firmware to acknowledge the triggered interrupts. When an interrupt bit in this register is read as 1, it indicates the corresponding interrupt has occurred and is waiting for service. Otherwise, no such interrupt is pending for service. When the firmware writes 1 to an interrupt bit, the corresponding interrupt will be acknowledged and its interrupt status bit will be cleared as 0. Writing 0 to an interrupt bit has no effect on it.

12.5.15 QMU Interrupt Mask Register (QIMR)

00000C04 QIMR QMU Interrupt Mask Register 1E03FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				TxE_EP_ERR_INTM	TxQ_ERR_INTM	RxE_EP_ERR_INTM	RxQ_ERR_INTM	RxBEQ_ERR_INTM				RBEQ_DEQ_INTM	RBEQ_EMPTY_INTM	RBEQ_DONE_INTM	RxQ_EMPTY_INTM	TxQ_EMPTY_INTM
Type				RW	RW	RW	RW	RW				RW	RW	RW	RW	RW
Reset				1	1	1	1	0				0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_DONE_INTM	R6Q_DONE_INTM	R5Q_DONE_INTM	R4Q_DONE_INTM	R3Q_DONE_INTM	R2Q_DONE_INTM	R1Q_DONE_INTM	R0Q_DONE_INTM	T7Q_DONE_INTM	T6Q_DONE_INTM	T5Q_DONE_INTM	T4Q_DONE_INTM	T3Q_DONE_INTM	T2Q_DONE_INTM	T1Q_DONE_INTM	T0Q_DONE_INTM
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
28	TxE_EP_ERR_INTM	TxE_EP_ERR_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
27	TxQ_ERR_INTM	TxQ_ERR_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
26	RxE_EP_ERR_INTM	RxE_EP_ERR_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
25	RxQ_ERR_INTM	RxQ_ERR_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
24	RxBEQ_ERR_INTM	RxBEQ_ERR_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
20	RBEQ_DEQ_INTM	RBEQ_DEQ_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
19	RBEQ_EMPTY_INTM	RBEQ_EMPTY_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
18	RBEQ_DONE_INTM	RBEQ_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
17	RxQ_EMPTY_INTM	RxQ_EMPTY_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.

16	TxQ_EMPTY_INTM	TxQ_EMPTY_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
15	R7Q_DONE_INTM	R7Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
14	R6Q_DONE_INTM	R6Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
13	R5Q_DONE_INTM	R5Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
12	R4Q_DONE_INTM	R4Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
11	R3Q_DONE_INTM	R3Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
10	R2Q_DONE_INTM	R2Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
9	R1Q_DONE_INTM	R1Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
8	R0Q_DONE_INTM	R0Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
7	T7Q_DONE_INTM	T7Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
6	T6Q_DONE_INTM	T6Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
5	T5Q_DONE_INTM	T5Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
4	T4Q_DONE_INTM	T4Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the

		corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
3	T3Q_DONE_INTM	T3Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
2	T2Q_DONE_INTM	T2Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
1	T1Q_DONE_INTM	T1Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.
0	T0Q_DONE_INTM	T0Q_DONE_INT Interrupt Mask Bits 31-0 in this register are defined as the individual interrupt mask bits for the corresponding interrupts defined in the Interrupt Status and Acknowledgement Register. 1'b0: interrupt enabled. 1'b1: interrupt disabled.

12.5.16 QMU Interrupt Mask Clear Register (QIMCR)

00000C08 QIMCR QMU Interrupt Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				TxEP_ERR_INTMC	TxQ_ERR_INTMC	RxEP_ERR_INTMC	RxQ_ERR_INTMC	RxBEQ_ERR_INTMC				RBEQ_DEQ_INTMC	RBEQ_EMPTY_INTMC	RBEQ_DONE_INTMC	RxQ_EMPTY_INTMC	TxQ_EMPTY_INTMC
Type				A0	A0	A0	A0	A0				A0	A0	A0	A0	A0
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_DONE_INTMC	R6Q_DONE_INTMC	R5Q_DONE_INTMC	R4Q_DONE_INTMC	R3Q_DONE_INTMC	R2Q_DONE_INTMC	R1Q_DONE_INTMC	R0Q_DONE_INTMC	T7Q_DONE_INTMC	T6Q_DONE_INTMC	T5Q_DONE_INTMC	T4Q_DONE_INTMC	T3Q_DONE_INTMC	T2Q_DONE_INTMC	T1Q_DONE_INTMC	T0Q_DONE_INTMC
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	TxEP_ERR_INTMC	TxEP_ERR_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
27	TxQ_ERR_INTMC	TxQ_ERR_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
26	RxEP_ERR_INTMC	RxEP_ERR_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register.

		1'b0: No effect 1'b1: Clear interrupt mask bit
25	RxQ_ERR_INTMC	RxQ_ERR_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
24	RxBEQ_ERR_INTMC	RxBEQ_ERR_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
20	RBEQ_DEQ_INTMC	RBEQ_DEQ_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
19	RBEQ_EMPTY_INTMC	RBEQ_EMPTY_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
18	RBEQ_DONE_INTMC	RBEQ_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
17	RxQ_EMPTY_INTMC	RxQ_EMPTY_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
16	TxQ_EMPTY_INTMC	TxQ_EMPTY_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
15	R7Q_DONE_INTMC	R7Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
14	R6Q_DONE_INTMC	R6Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
13	R5Q_DONE_INTMC	R5Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
12	R4Q_DONE_INTMC	R4Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
11	R3Q_DONE_INTMC	R3Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit

10	R2Q_DONE_INTMC	R2Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
9	R1Q_DONE_INTMC	R1Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
8	R0Q_DONE_INTMC	R0Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
7	T7Q_DONE_INTMC	T7Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
6	T6Q_DONE_INTMC	T6Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
5	T5Q_DONE_INTMC	T5Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
4	T4Q_DONE_INTMC	T4Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
3	T3Q_DONE_INTMC	T3Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
2	T2Q_DONE_INTMC	T2Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
1	T1Q_DONE_INTMC	T1Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit
0	T0Q_DONE_INTMC	T0Q_DONE_INT Interrupt Mask Clear Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Clear interrupt mask bit

12.5.17 QMU Interrupt Mask Set Register (QIMSR)

00000C0C QIMSR QMU Interrupt Mask Set Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				TxEP_ERR_INTMS	TxQ_ERR_INTMS	RxEP_ERR_INTMS	RxQ_ERR_INTMS	RxBEQ_ERR_INTMS				RBEQ_DEQ_INTMS	RBEQ_EMPTY_INTMS	RBEQ_DONE_INTMS	RxQ_EMPTY_INTMS	TxQ_EMPTY_INTMS
Type				A0	A0	A0	A0	A0				A0	A0	A0	A0	A0
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_DONE_INTMS	R6Q_DONE_INTMS	R5Q_DONE_INTMS	R4Q_DONE_INTMS	R3Q_DONE_INTMS	R2Q_DONE_INTMS	R1Q_DONE_INTMS	R0Q_DONE_INTMS	T7Q_DONE_INTMS	T6Q_DONE_INTMS	T5Q_DONE_INTMS	T4Q_DONE_INTMS	T3Q_DONE_INTMS	T2Q_DONE_INTMS	T1Q_DONE_INTMS	T0Q_DONE_INTMS
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28	TxEP_ERR_INTMS	TxEP_ERR_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
27	TxQ_ERR_INTMS	TxQ_ERR_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
26	RxEP_ERR_INTMS	RxEP_ERR_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
25	RxQ_ERR_INTMS	RxQ_ERR_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
24	RxBEQ_ERR_INTMS	RxBEQ_ERR_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
20	RBEQ_DEQ_INTMS	RBEQ_DEQ_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
19	RBEQ_EMPTY_INTMS	RBEQ_EMPTY_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
18	RBEQ_DONE_INTMS	RBEQ_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
17	RxQ_EMPTY_INTMS	RxQ_EMPTY_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect

		1'b1: Set interrupt mask bit
16	TxQ_EMPTY_INTMS	TxQ_EMPTY_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
15	R7Q_DONE_INTMS	R7Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
14	R6Q_DONE_INTMS	R6Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
13	R5Q_DONE_INTMS	R5Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
12	R4Q_DONE_INTMS	R4Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
11	R3Q_DONE_INTMS	R3Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
10	R2Q_DONE_INTMS	R2Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
9	R1Q_DONE_INTMS	R1Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
8	R0Q_DONE_INTMS	R0Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
7	T7Q_DONE_INTMS	T7Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
6	T6Q_DONE_INTMS	T6Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
5	T5Q_DONE_INTMS	T5Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
4	T4Q_DONE_INTMS	T4Q_DONE_INT Interrupt Mask Set

		Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
3	T3Q_DONE_INTMS	T3Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
2	T2Q_DONE_INTMS	T2Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
1	T1Q_DONE_INTMS	T1Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit
0	T0Q_DONE_INTMS	T0Q_DONE_INT Interrupt Mask Set Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. 1'b0: No effect 1'b1: Set interrupt mask bit

12.5.18 Device Software Interrupt Command Register (DSICR)

00000C20 DSICR Device Software Interrupt Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DSICR															
Type	A0															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	DSICR	Device Firmware Interrupt Status Register Interrupt request is fired to host when writing 1 to this register and the corresponding bit is set in host side's SISAR. HIF Event.

12.5.19 GPD Done Interrupt on GPD IOC bit Disable Register (IOCDISR)

00000C30 IOCDISR GPD Done Interrupt on GPD IOC bit Disable Register 0000FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_IOC_DIS	R6Q_IOC_DIS	R5Q_IOC_DIS	R4Q_IOC_DIS	R3Q_IOC_DIS	R2Q_IOC_DIS	R1Q_IOC_DIS	R0Q_IOC_DIS	T7Q_IOC_DIS	T6Q_IOC_DIS	T5Q_IOC_DIS	T4Q_IOC_DIS	T3Q_IOC_DIS	T2Q_IOC_DIS	T1Q_IOC_DIS	T0Q_IOC_DIS
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
15	R7Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
14	R6Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
13	R5Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
12	R4Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
11	R3Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
10	R2Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
9	R1Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function

8	R0Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
7	T7Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
6	T6Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
5	T5Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
4	T4Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
3	T3Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
2	T2Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
1	T1Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done. Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR. 1'b0: Enable IOC function. 1'b1: Disable IOC function
0	T0Q_IOC_DIS	GPD Done Interrupt on GPD IOC bit Disable Register

Writing 1 to this register, the corresponding queue always issue interrupt event when GPD is done.
Writing 0 to this register, the corresponding queue will issue interrupt event base on GPD IOC bit. If current GPD IOC = 1, GPD done interrupt event will be issued and latched into QISAR.
1'b0: Enable IOC function.
1'b1: Disable IOC function

12.5.20 QMU Hardware Version Register (QMU_HWVER)

00000C40 QMU_HWVER QMU Hardware Version Register 00002402

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																WIMAX
Type																RO
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXQ_NUM				TXQ_NUM				HWVER							
Type	RO				RO				RO							
Reset	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0

Bit(s)	Name	Description
16	WIMAX	Wimax specific feature define 1'b1: Wimax HIF
15:12	RXQ_NUM	RX Queue Number
11:8	TXQ_NUM	TX Queue Number
7:0	HWVER	Hardware Version Code

12.5.21 TX Queue Empty Indication Register (TQEMIR)

00000C60 TQEMIR TX Queue Empty Indication Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7Q_EMPTY	T6Q_EMPTY	T5Q_EMPTY	T4Q_EMPTY	T3Q_EMPTY	T2Q_EMPTY	T1Q_EMPTY	T0Q_EMPTY
Type									W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	T7Q_EMPTY	T7 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty

6	T6Q_EMPTY	T6 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
5	T5Q_EMPTY	T5 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
4	T4Q_EMPTY	T4 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
3	T3Q_EMPTY	T3 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
2	T2Q_EMPTY	T2 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
1	T1Q_EMPTY	T1 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
0	T0Q_EMPTY	T0 Queue Empty Register Writing 1 to this register, the corresponding TX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty

12.5.22 TX Queue Empty Indication Mask Register (TQEMIMR)

00000C64 TQEMIMR TX Queue Empty Indication Mask Register 000000FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7Q_EMPTY_IMR	T6Q_EMPTY_IMR	T5Q_EMPTY_IMR	T4Q_EMPTY_IMR	T3Q_EMPTY_IMR	T2Q_EMPTY_IMR	T1Q_EMPTY_IMR	T0Q_EMPTY_IMR
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1

Bit(s)	Name	Description
7	T7Q_EMPTY_IMR	T7 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.

6	T6Q_EMPTY_IMR	T6 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
5	T5Q_EMPTY_IMR	T5 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
4	T4Q_EMPTY_IMR	T4 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
3	T3Q_EMPTY_IMR	T3 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
2	T2Q_EMPTY_IMR	T2 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
1	T1Q_EMPTY_IMR	T1 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
0	T0Q_EMPTY_IMR	T0 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding TX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.

12.5.23 TX Queue Empty Indication Mask Clear Register (TQEMIMCR)

00000C68 TQEMIMCR TX Queue Empty Indication Mask Clear Register 00000000

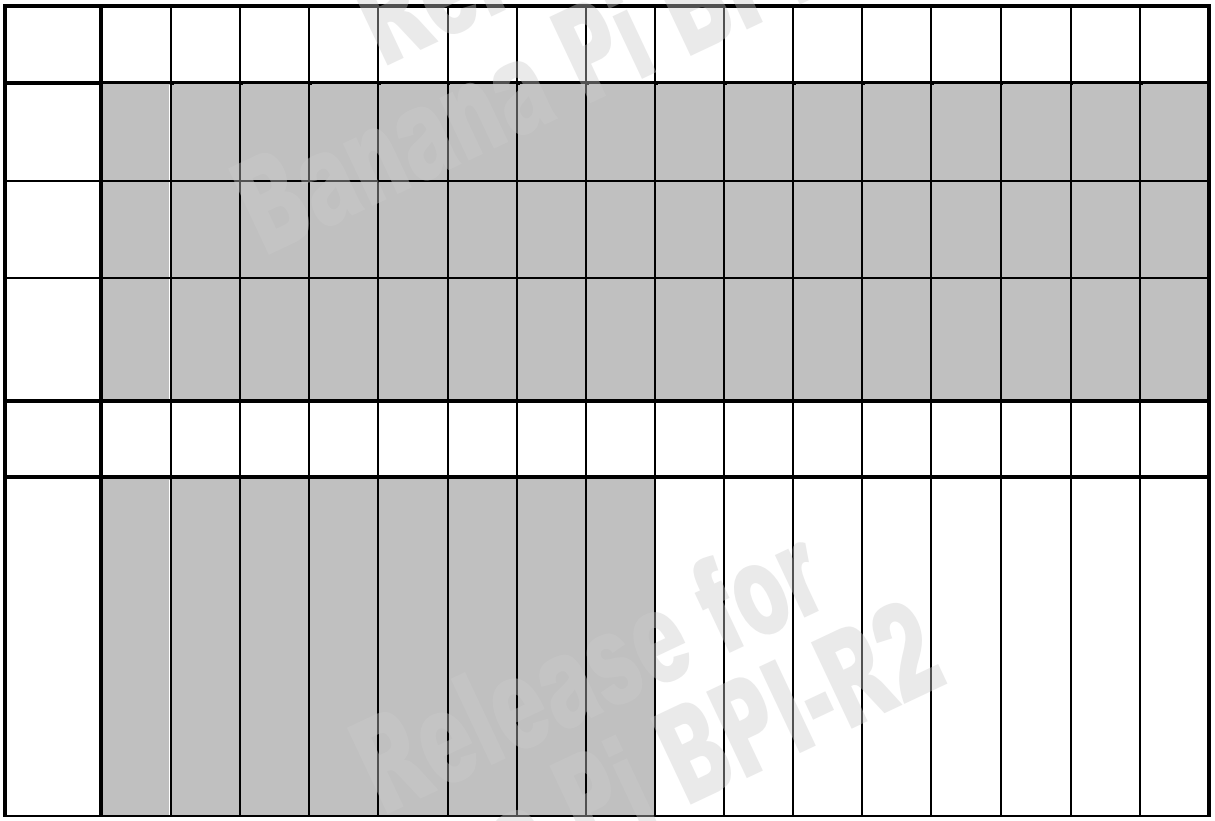
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7Q_EMPTY_IMCR	T6Q_EMPTY_IMCR	T5Q_EMPTY_IMCR	T4Q_EMPTY_IMCR	T3Q_EMPTY_IMCR	T2Q_EMPTY_IMCR	T1Q_EMPTY_IMCR	T0Q_EMPTY_IMCR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0

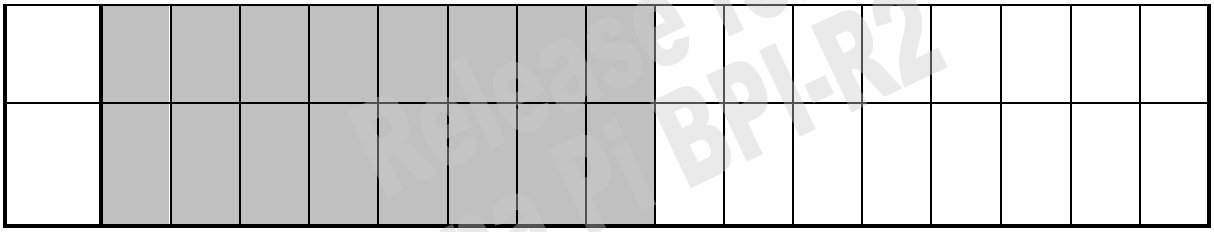
Bit(s)	Name	Description
7	T7Q_EMPTY_IMCR	T7 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
6	T6Q_EMPTY_IMCR	T6 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.

5	T5Q_EMPTY_IMCR	T5 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
4	T4Q_EMPTY_IMCR	T4 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
3	T3Q_EMPTY_IMCR	T3 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
2	T2Q_EMPTY_IMCR	T2 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
1	T1Q_EMPTY_IMCR	T1 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.
0	T0Q_EMPTY_IMCR	T0 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding TX queue indication mask bit is cleared.

12.5.24 TX Queue Empty Indication Mask Set Register (TQEMIMSR)

TX Queue Empty Indication Mask Set Register





	Description
	T7 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding TX queue indication mask bit is set.
	T6 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding TX queue indication mask bit is set.
	T5 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding TX queue indication mask bit is set.
	T4 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding TX queue indication mask bit is set.

T3 Queue Empty Indication Mask Set Register
Writing 1 to this register, the corresponding TX queue indication mask bit is set.

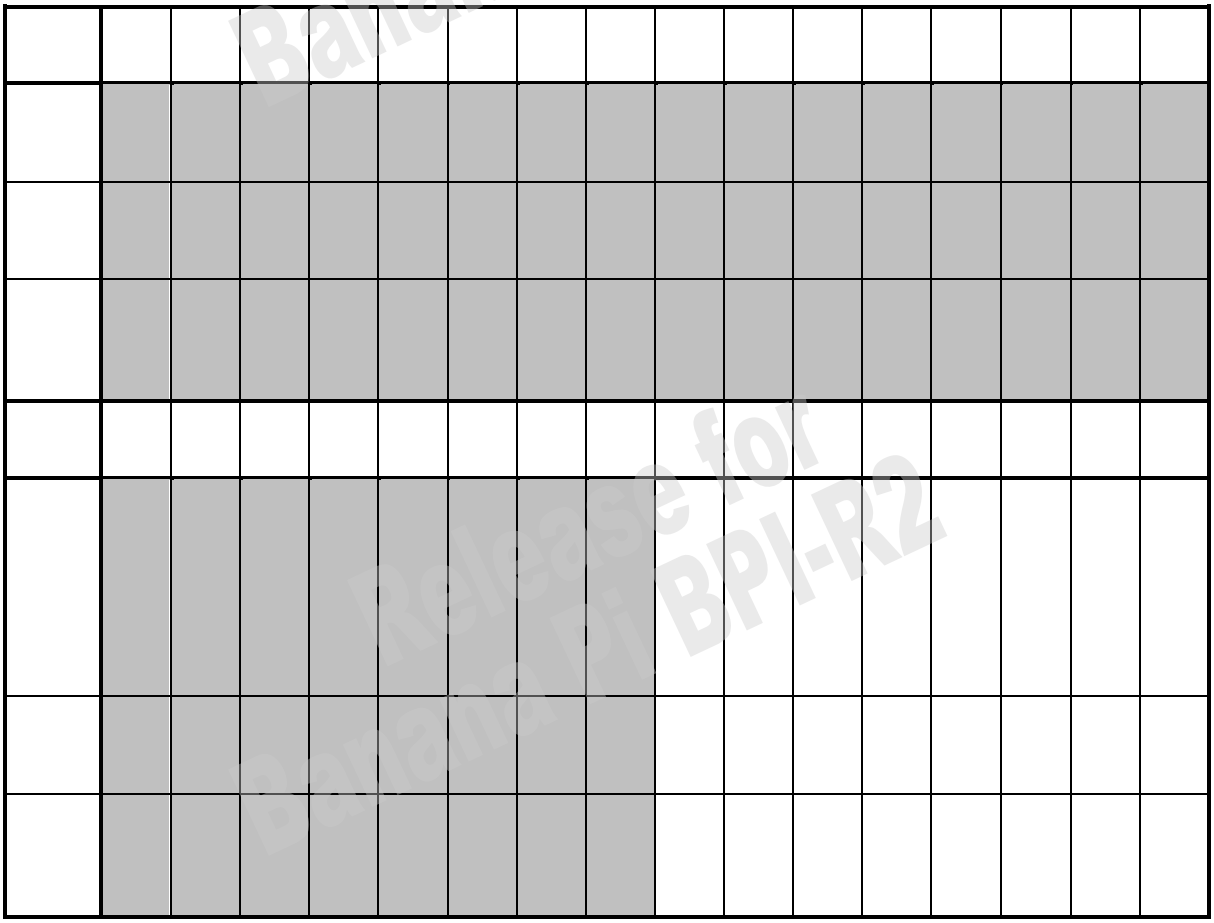
T2 Queue Empty Indication Mask Set Register
Writing 1 to this register, the corresponding TX queue indication mask bit is set.

T1 Queue Empty Indication Mask Set Register
Writing 1 to this register, the corresponding TX queue indication mask bit is set.

T0 Queue Empty Indication Mask Set Register
Writing 1 to this register, the corresponding TX queue indication mask bit is set.

12.5.25 RX Queue Empty Indication Register (RQEMIR)

RX Queue Empty
Indication Register



	Description
	R7 Queue Empty Register Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked. 1'b0: The corresponding queue has not read a null GPD yet 1'b1: The corresponding Queue has become empty
	R6 Queue Empty Register Writing 1 to this register, the corresponding RX queue empty generated from USB MAC

should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R5 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R4 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R3 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R2 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R1 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

R0 Queue Empty Register

Writing 1 to this register, the corresponding RX queue empty generated from USB MAC should be masked.
1'b0: The corresponding queue has not read a null GPD yet
1'b1: The corresponding Queue has become empty

12.5.26 RX Queue Empty Indication Mask Register (RQEMIMR)

00000C74 RQEMIMR RX Queue Empty Indication Mask Register 000000FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									R7Q_EMPTY_IMR	R6Q_EMPTY_IMR	R5Q_EMPTY_IMR	R4Q_EMPTY_IMR	R3Q_EMPTY_IMR	R2Q_EMPTY_IMR	R1Q_EMPTY_IMR	R0Q_EMPTY_IMR
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1

Bit(s)	Name	Description
7	R7Q_EMPTY_IMR	R7 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
6	R6Q_EMPTY_IMR	R6 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
5	R5Q_EMPTY_IMR	R5 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
4	R4Q_EMPTY_IMR	R4 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
3	R3Q_EMPTY_IMR	R3 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
2	R2Q_EMPTY_IMR	R2 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
1	R1Q_EMPTY_IMR	R1 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.
0	R0Q_EMPTY_IMR	R0 Queue Empty Indication Mask Register Writing 1 to this register, the corresponding RX queue indication generated from USB MAC should be masked. 1'b0: The corresponding queue indication from QMU should be unmasked. 1'b1: The corresponding queue indication from QMU should be masked.

12.5.27 RX Queue Empty Indication Mask Clear Register (RQEMIMCR)

00000C78 RQEMIMCR RX Queue Empty Indication Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									R7Q_EMPTY_IMCR	R6Q_EMPTY_IMCR	R5Q_EMPTY_IMCR	R4Q_EMPTY_IMCR	R3Q_EMPTY_IMCR	R2Q_EMPTY_IMCR	R1Q_EMPTY_IMCR	R0Q_EMPTY_IMCR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	R7Q_EMPTY_IMCR	R7 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
6	R6Q_EMPTY_IMCR	R6 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
5	R5Q_EMPTY_IMCR	R5 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
4	R4Q_EMPTY_IMCR	R4 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
3	R3Q_EMPTY_IMCR	R3 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
2	R2Q_EMPTY_IMCR	R2 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
1	R1Q_EMPTY_IMCR	R1 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.
0	R0Q_EMPTY_IMCR	R0 Queue Empty Indication Mask Clear Register Writing 1 to this register, the corresponding RX queue indication mask bit is cleared.

12.5.28 RX Queue Empty Indication Mask Set Register (RQEMIMSR)

00000C7C RQEMIMSR RX Queue Empty Indication Mask Set Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									R7Q_EMPTY_IMSR	R6Q_EMPTY_IMSR	R5Q_EMPTY_IMSR	R4Q_EMPTY_IMSR	R3Q_EMPTY_IMSR	R2Q_EMPTY_IMSR	R1Q_EMPTY_IMSR	R0Q_EMPTY_IMSR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	R7Q_EMPTY_IMSR	R7 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.

6	R6Q_EMPTY_IMSR	R6 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
5	R5Q_EMPTY_IMSR	R5 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
4	R4Q_EMPTY_IMSR	R4 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
3	R3Q_EMPTY_IMSR	R3 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
2	R2Q_EMPTY_IMSR	R2 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
1	R1Q_EMPTY_IMSR	R1 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.
0	R0Q_EMPTY_IMSR	R0 Queue Empty Indication Mask Set Register Writing 1 to this register, the corresponding RX queue indication mask bit is set.

12.5.29 RX Queue Error Indication Register (RQEIR)

00000C90 RQEIR RX Queue Error Indication Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R7Q_ZLP_ERR	R6Q_ZLP_ERR	R5Q_ZLP_ERR	R4Q_ZLP_ERR	R3Q_ZLP_ERR	R2Q_ZLP_ERR	R1Q_ZLP_ERR	R0Q_ZLP_ERR	R7Q_HDR_ERR	R6Q_HDR_ERR	R5Q_HDR_ERR	R4Q_HDR_ERR	R3Q_HDR_ERR	R2Q_HDR_ERR	R1Q_HDR_ERR	R0Q_HDR_ERR
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_LEN_ERR	R6Q_LEN_ERR	R5Q_LEN_ERR	R4Q_LEN_ERR	R3Q_LEN_ERR	R2Q_LEN_ERR	R1Q_LEN_ERR	R0Q_LEN_ERR	R7Q_GPD_CSER	R6Q_GPD_CSER	R5Q_GPD_CSER	R4Q_GPD_CSER	R3Q_GPD_CSER	R2Q_GPD_CSER	R1Q_GPD_CSER	R0Q_GPD_CSER
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	R7Q_ZLP_ERR	R7 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
30	R6Q_ZLP_ERR	R6 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
29	R5Q_ZLP_ERR	R5 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
28	R4Q_ZLP_ERR	R4 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because

		incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
27	R3Q_ZLP_ERR	R3 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
26	R2Q_ZLP_ERR	R2 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
25	R1Q_ZLP_ERR	R1 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
24	R0Q_ZLP_ERR	R0 Queue ZLP Error Indication This bit is used to indicate that an unexpected ZLP packet was received. Note: ZLP_ERR will not cause HW to stop, unlike all the other errors. This is because incoming packets could still be valid and should received. 1'b0: No error occurs 1'b1: Unexpected ZLP error occurs.
23	R7Q_HDR_ERR	R7 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
22	R6Q_HDR_ERR	R6 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
21	R5Q_HDR_ERR	R5 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
20	R4Q_HDR_ERR	R4 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
19	R3Q_HDR_ERR	R3 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
18	R2Q_HDR_ERR	R2 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.

17	R1Q_HDR_ERR	R1 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
16	R0Q_HDR_ERR	R0 Queue Header Error Indication (wimax only) This bit is used to indicate whether host send an RxQ SDU and the header checksum is not correct, or the type indicate is wrong, or the SDU length was over received USB packet length but not received max packet, or SDU length was zero. 1'b0: No header error occurs. 1'b1: Header error occurs.
15	R7Q_LEN_ERR	R7 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
14	R6Q_LEN_ERR	R6 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
13	R5Q_LEN_ERR	R5 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
12	R4Q_LEN_ERR	R4 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
11	R3Q_LEN_ERR	R3 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
10	R2Q_LEN_ERR	R2 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
9	R1Q_LEN_ERR	R1 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic). This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
8	R0Q_LEN_ERR	R0 VI Queue Length Error If the received data put into the RX GPD is over the Data Buffer Length of the RX GPD, this bit will assert. (generic).

		This bit is used to indicate when host sends an RxQ SDU and length is over dbuf size in QCR1. (wimax) 1'b0: No length error occurs. 1'b1: length error occurs.
7	R7Q_GPD_CSERR	R7 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
6	R6Q_GPD_CSERR	R6 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
5	R5Q_GPD_CSERR	R5 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
4	R4Q_GPD_CSERR	R4 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
3	R3Q_GPD_CSERR	R3 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
2	R2Q_GPD_CSERR	R2 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
1	R1Q_GPD_CSERR	R1 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.
0	R0Q_GPD_CSERR	R0 Queue Generic Packet Descriptor Checksum Error This bit is used to indicate whether a Generic Packet Descriptor checksum error occurred in the specific RX Queue. 1'b0: No RQ-GPD checksum error occurred. 1'b1: RQ-GPD checksum error occurred.

12.5.30 RX Queue Error Indication Mask Register (RQEIMR)

00000C94 RQEIMR RX Queue Error Indication Mask Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R7Q_ZLP_ERR_IMR	R6Q_ZLP_ERR_IMR	R5Q_ZLP_ERR_IMR	R4Q_ZLP_ERR_IMR	R3Q_ZLP_ERR_IMR	R2Q_ZLP_ERR_IMR	R1Q_ZLP_ERR_IMR	R0Q_ZLP_ERR_IMR	R7Q_HDR_ERR_IMR	R6Q_HDR_ERR_IMR	R5Q_HDR_ERR_IMR	R4Q_HDR_ERR_IMR	R3Q_HDR_ERR_IMR	R2Q_HDR_ERR_IMR	R1Q_HDR_ERR_IMR	R0Q_HDR_ERR_IMR
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	R7Q_	R6Q_L	R5Q_L	R4Q_L	R3Q_L	R2Q_L	R1Q_L	R0Q_L	R7Q_	R6Q_	R5Q_	R4Q_	R3Q_	R2Q_	R1Q_	R0Q_

	LEN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	EN_ERR_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR	GPD_CSER_IMR
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31	R7Q_ZLP_ERR_IMR	R7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
30	R6Q_ZLP_ERR_IMR	R6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
29	R5Q_ZLP_ERR_IMR	R5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
28	R4Q_ZLP_ERR_IMR	R4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
27	R3Q_ZLP_ERR_IMR	R. Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
26	R2Q_ZLP_ERR_IMR	R2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
25	R1Q_ZLP_ERR_IMR	R1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
24	R0Q_ZLP_ERR_IMR	R0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
23	R7Q_HDR_ERR_IMR	R7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
22	R6Q_HDR_ERR_IMR	R6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
21	R5Q_HDR_ERR_IMR	R5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.

20	R4Q_HDR_ERR_IMR	R4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
19	R3Q_HDR_ERR_IMR	R3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
18	R2Q_HDR_ERR_IMR	R2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
17	R1Q_HDR_ERR_IMR	R1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
16	R0Q_HDR_ERR_IMR	R0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
15	R7Q_LEN_ERR_IMR	R7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
14	R6Q_LEN_ERR_IMR	R6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
13	R5Q_LEN_ERR_IMR	R5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
12	R4Q_LEN_ERR_IMR	R4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
11	R3Q_LEN_ERR_IMR	R3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
10	R2Q_LEN_ERR_IMR	R2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
9	R1Q_LEN_ERR_IMR	R1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
8	R0Q_LEN_ERR_IMR	R0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from

		QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
7	R7Q_GPD_CSERR_IMR	R7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
6	R6Q_GPD_CSERR_IMR	R6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
5	R5Q_GPD_CSERR_IMR	R5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
4	R4Q_GPD_CSERR_IMR	R4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
3	R3Q_GPD_CSERR_IMR	R3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
2	R2Q_GPD_CSERR_IMR	R2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
1	R1Q_GPD_CSERR_IMR	R1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.
0	R0Q_GPD_CSERR_IMR	R0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding RX queue error indication generated from QMU should be masked. 1'b0: The corresponding RX queue error indication from QMU should be unmasked. 1'b1: The corresponding RX queue error indication from QMU should be masked.

12.5.31 RX Queue Error Indication Mask Clear Register (RQEIMCR)

00000C98 RQEIMCR RX Queue Error Indication Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	R7Q_ZLP_ERR_IMCR	R6Q_ZLP_ERR_IMCR	R5Q_ZLP_ERR_IMCR	R4Q_ZLP_ERR_IMCR	R3Q_ZLP_ERR_IMCR	R2Q_ZLP_ERR_IMCR	R1Q_ZLP_ERR_IMCR	R0Q_ZLP_ERR_IMCR	R7Q_HDR_ERR_IMCR	R6Q_HDR_ERR_IMCR	R5Q_HDR_ERR_IMCR	R4Q_HDR_ERR_IMCR	R3Q_HDR_ERR_IMCR	R2Q_HDR_ERR_IMCR	R1Q_HDR_ERR_IMCR	R0Q_HDR_ERR_IMCR
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	R7Q_LEN_ERR_IMCR	R6Q_LEN_ERR_IMCR	R5Q_LEN_ERR_IMCR	R4Q_LEN_ERR_IMCR	R3Q_LEN_ERR_IMCR	R2Q_LEN_ERR_IMCR	R1Q_LEN_ERR_IMCR	R0Q_LEN_ERR_IMCR	R7Q_GPD_CSER_IMCR	R6Q_GPD_CSER_IMCR	R5Q_GPD_CSER_IMCR	R4Q_GPD_CSER_IMCR	R3Q_GPD_CSER_IMCR	R2Q_GPD_CSER_IMCR	R1Q_GPD_CSER_IMCR	R0Q_GPD_CSER_IMCR
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s	Name	Description
31	R7Q_ZLP_ERR_IMCR	R7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
30	R6Q_ZLP_ERR_IMCR	R6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
29	R5Q_ZLP_ERR_IMCR	R5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
28	R4Q_ZLP_ERR_IMCR	R4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
27	R3Q_ZLP_ERR_IMCR	R3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
26	R2Q_ZLP_ERR_IMCR	R2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
25	R1Q_ZLP_ERR_IMCR	R1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
24	R0Q_ZLP_ERR_IMCR	R0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.

1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

23	R7Q_HDR_ERR_IMCR	R7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
22	R6Q_HDR_ERR_IMCR	R6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
21	R5Q_HDR_ERR_IMCR	R5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
20	R4Q_HDR_ERR_IMCR	R4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
19	R3Q_HDR_ERR_IMCR	R3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
18	R2Q_HDR_ERR_IMCR	R2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
17	R1Q_HDR_ERR_IMCR	RX Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
16	R0Q_HDR_ERR_IMCR	R0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
15	R7Q_LEN_ERR_IMCR	R7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

14	R6Q_LEN_ERR_IMCR	R6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
13	R5Q_LEN_ERR_IMCR	R5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
12	R4Q_LEN_ERR_IMCR	R4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
11	R3Q_LEN_ERR_IMCR	R3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
10	R2Q_LEN_ERR_IMCR	R2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
9	R1Q_LEN_ERR_IMCR	R1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
8	R0Q_LEN_ERR_IMCR	R0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
7	R7Q_GPD_CSERR_IMCR	R7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
6	R6Q_GPD_CSERR_IMCR	R6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
5	R5Q_GPD_CSERR	R5 Queue Error Indication Mask Clear Register

	_IMCR	Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
4	R4Q_GPD_CSERR_IMCR	R4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
3	R3Q_GPD_CSERR_IMCR	R3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
2	R2Q_GPD_CSERR_IMCR	R2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
1	R1Q_GPD_CSERR_IMCR	R1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
0	R0Q_GPD_CSERR_IMCR	R0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

12.5.32 RX Queue Error Indication Mask Set Register (RQEIMSR)

00000C9C	RQEIMSR	RX Queue Error Indication Mask Set Register	00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Na	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Ty	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Re	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	1	1	1	1	1	1	9	8	7	6	5	4	3	2	1	0
Na	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Ty	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Re	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	R7Q_ZLP_ERR_IMSR	R7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
30	R6Q_ZLP_ERR_IMSR	R6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
29	R5Q_ZLP_ERR_IMSR	R5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
28	R4Q_ZLP_ERR_IMSR	R4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

27	R3Q_ZLP_ERR_IMSR	R3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
26	R2Q_ZLP_ERR_IMSR	R2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
25	R1Q_ZLP_ERR_IMSR	R1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
24	R0Q_ZLP_ERR_IMSR	R0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
23	R7Q_HDR_ERR_IMSR	R7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
22	R6Q_HDR_ERR_IMSR	R6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
21	R5Q_HDR_ERR_IMSR	R5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
20	R4Q_HDR_ERR_IMSR	R4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
19	R3Q_HDR_ERR_IMSR	R3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
18	R2Q_HDR_ERR_IMSR	R2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
17	R1Q_HDR_ERR_IMSR	R1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.

		1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
16	R0Q_HDR_ERR_IMSR	R0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
15	R7Q_LEN_ERR_IMSR	R7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
14	R6Q_LEN_ERR_IMSR	R6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
13	R5Q_LEN_ERR_IMSR	R5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
12	R4Q_LEN_ERR_IMSR	R4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
11	R3Q_LEN_ERR_IMSR	R3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
10	R2Q_LEN_ERR_IMSR	R2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
9	R1Q_LEN_ERR_IMSR	R1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
8	R0Q_LEN_ERR_IMSR	R0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
7	R7Q_GPD_CSERR_IMSR	R7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
6	R6Q_GPD_CSERR_IMS	R6 Queue Error Indication Mask Set Register

	R	Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
5	R5Q_GPD_CSERR_IMSR	R5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
4	R4Q_GPD_CSERR_IMSR	R4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
3	R3Q_GPD_CSERR_IMSR	R3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
2	R2Q_GPD_CSERR_IMSR	R2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
1	R1Q_GPD_CSERR_IMSR	R1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
0	R0Q_GPD_CSERR_IMSR	R0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

12.5.33 RX Endpoint Error Indication Register (REPEIR)

00000CA0 REPEIR RX Endpoint Error Indication Register 00000000

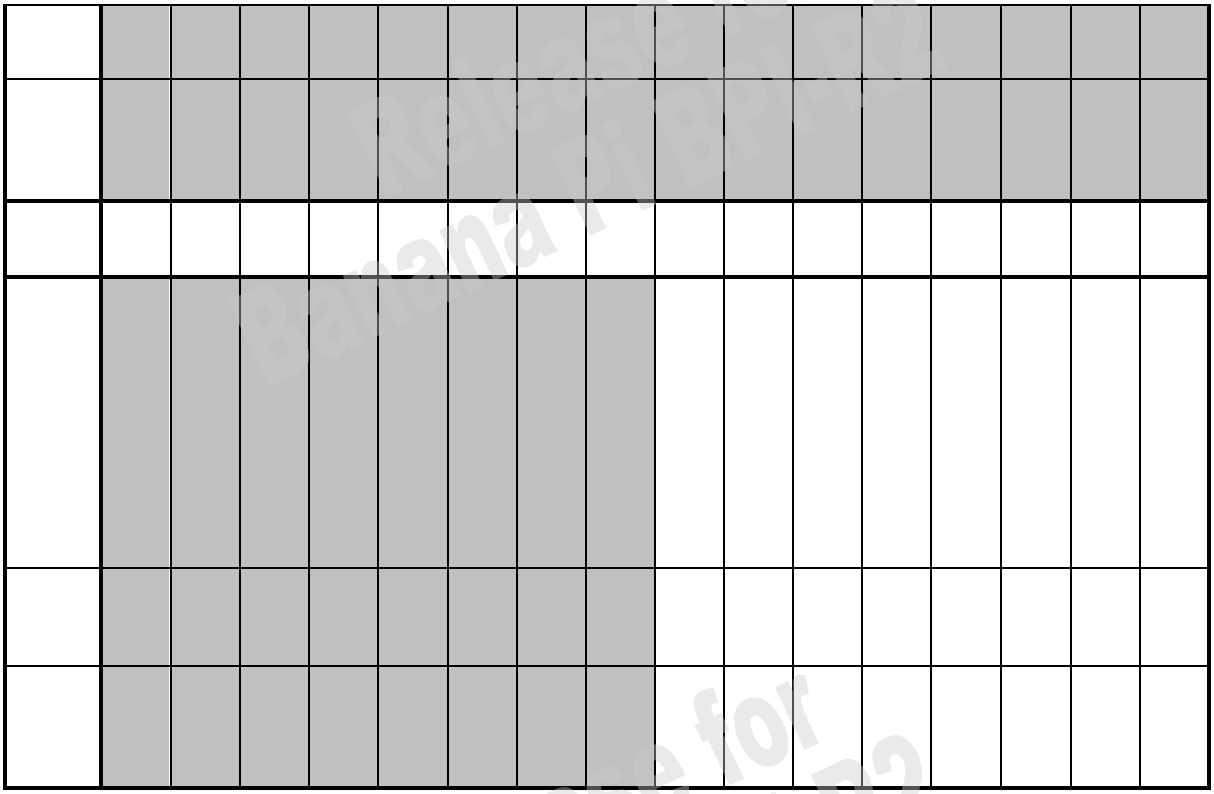
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									R7EP_ERR	R6EP_ERR	R5EP_ERR	R4EP_ERR	R3EP_ERR	R2EP_ERR	R1EP_ERR	R0EP_ERR
Type									W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	R7EP_ERR	R7 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to

		stop the corresponding queue. 1'b0: No error. 1'b1: Error.
6	R6EP_ERR	R6 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
5	R5EP_ERR	R5 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
4	R4EP_ERR	R4 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
3	R3EP_ERR	R3 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
2	R2EP_ERR	R2 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
1	R1EP_ERR	R1 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
0	R0EP_ERR	R0 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.

12.5.34 RX Endpoint Error Indication Mask Register (REPEIMR)

000 (((/ ¿	REPEIMR	RX Endpoint Error Indication Mask Register	00

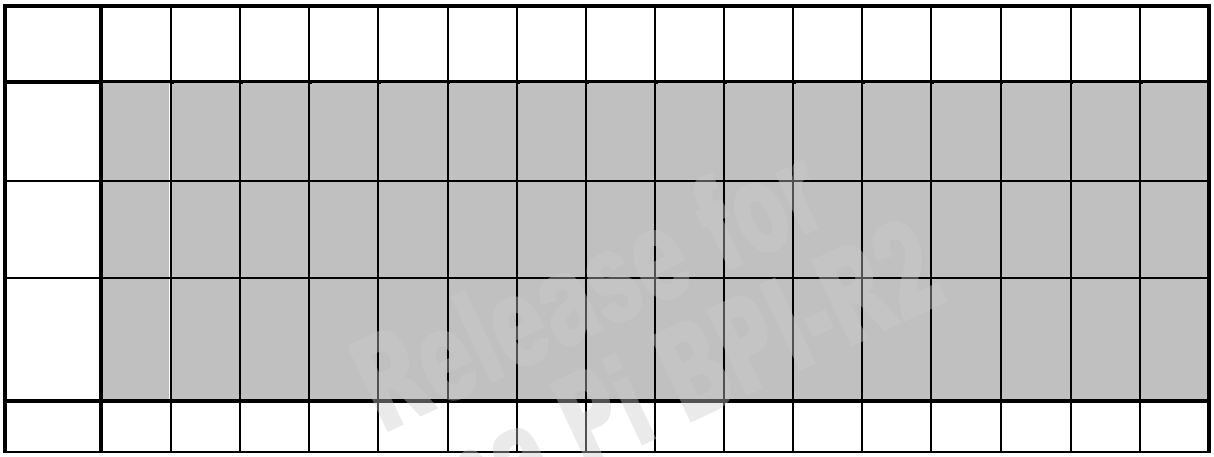


Name	Description
R7EP_ERR_IMR	R7 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R6EP_ERR_IMR	R6 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R5EP_ERR_IMR	R5 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R4EP_ERR_IMR	R4 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from

	EPMU should be masked.
R3EP_ERR_IMR	R3 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R2EP_ERR_IMR	R2 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R1EP_ERR_IMR	R1 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
R0EP_ERR_IMR	R0 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding RX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.

12.5.35 RX Endpoint Error Indication Mask Clear Register (REPEIMCR)

RX Endpoint Error Indication Mask Clear Register



Description

R7 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R6 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R5 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R4 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R3 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R2 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R1 Endpoint Error Indication Mask Clear Register

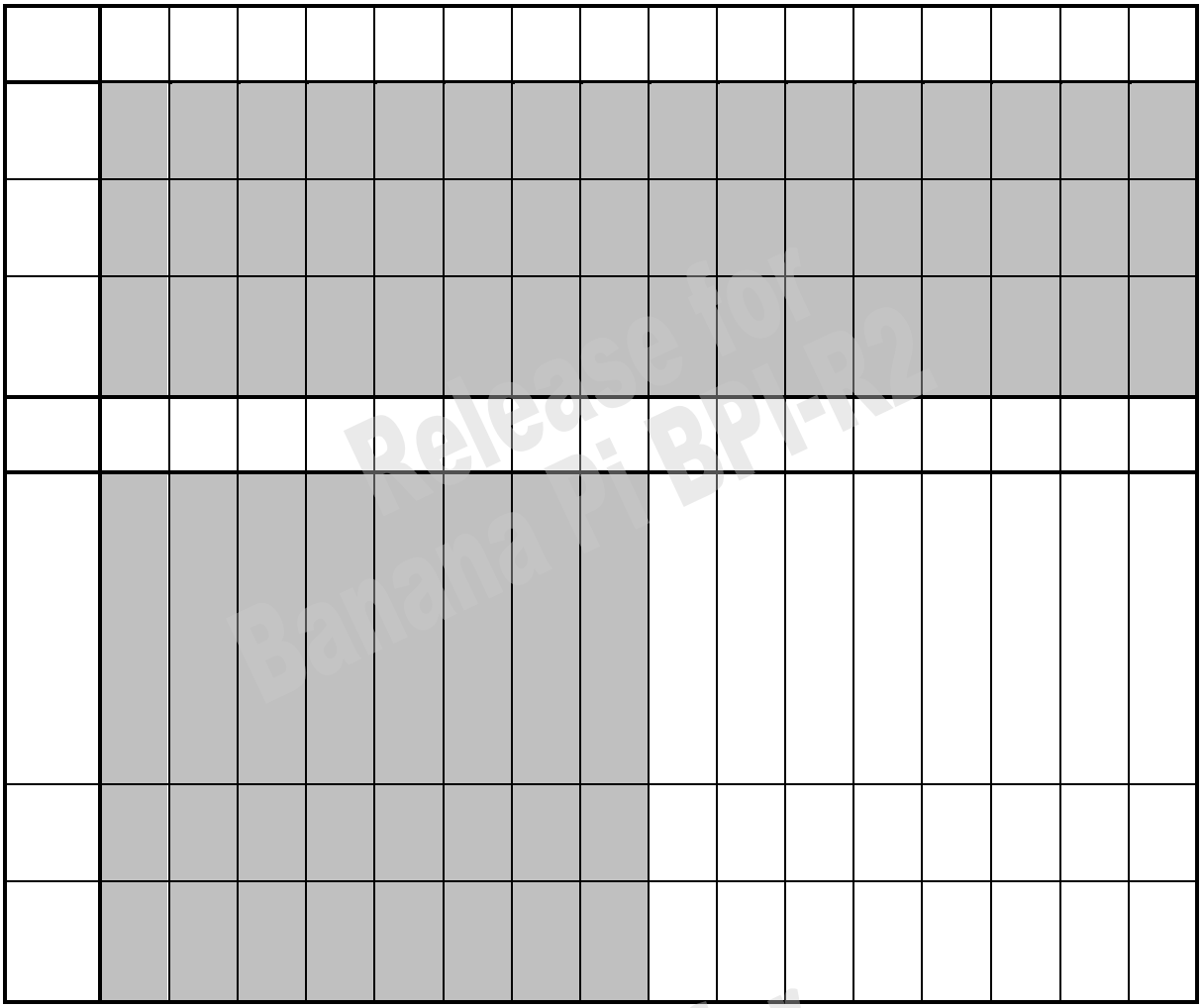
Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

R0 Endpoint Error Indication Mask Clear Register

Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

12.5.36 RX Endpoint Error Indication Mask Set Register (REPEIMSR)

000 REPEIMSR RX Endpoint Error Indication Mask Set Register 00

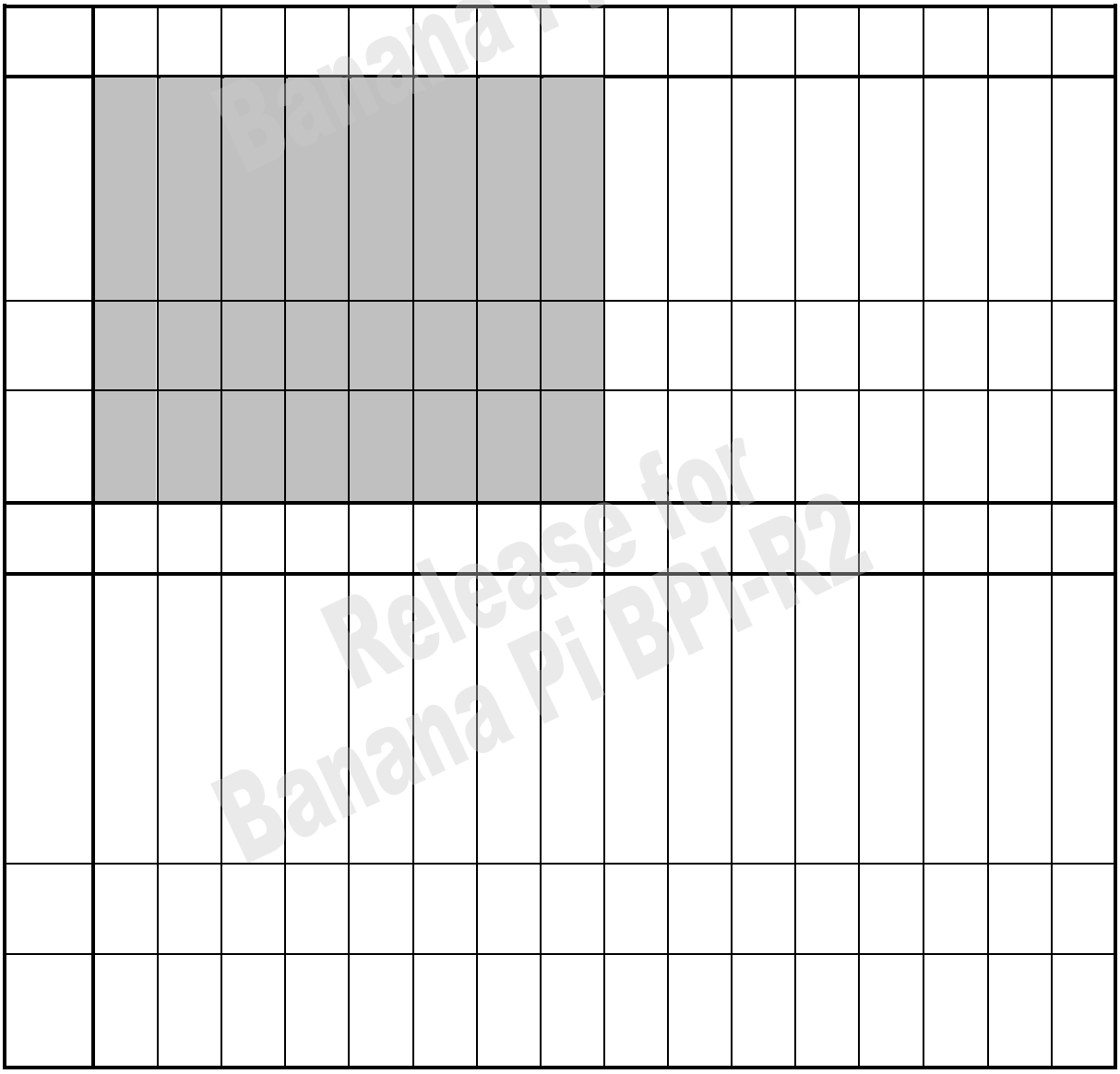


Name	Description
R7EP_ERR_IMS	R7 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask

R	bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R6EP_ERR_IMSR	R6 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R5EP_ERR_IMSR	R5 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R4EP_ERR_IMSR	R4 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R3EP_ERR_IMSR	R3 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R2EP_ERR_IMSR	R2 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R1EP_ERR_IMSR	R1 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
R0EP_ERR_IMSR	R0 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

12.5.37 TX Queue Error Indication Register (TQEIR)

TX Queue Error Indication Register



Description
T7 Queue Generic Packet Descriptor Checksum Error The Data Buffer Length and the Descriptor Extension Length are all zero. 1'b0: No length error. 1'b1: Length error.

T6 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T5 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T4 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T3 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T2 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T1 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.
1'b0: No length error.
1'b1: Length error.

T0 Queue Generic Packet Descriptor Checksum Error
The Data Buffer Length and the Descriptor Extension Length are all zero.

1'b0: No length error.
1'b1: Length error.

T7 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T6 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T5 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T4 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T3 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T2 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T1 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T0 Queue Generic Packet Descriptor Checksum Error
This bit is used to indicate the GPD checksum error in the TxQ.
1'b0: No GPD checksum error occurred in the TxQ.
1'b1: GPD checksum error occurred in the TxQ.

T7 Queue Buffer Descriptor Checksum Error
This bit is used to indicate the BD checksum error in the TxQ.
1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T6 Queue Buffer Descriptor Checksum Error
This bit is used to indicate the BD checksum error in the TxQ.
1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T5 Queue Buffer Descriptor Checksum Error
This bit is used to indicate the BD checksum error in the TxQ.
1'b0: No BD checksum error occurred in the TxQ.

1'b1: BD checksum error occurred in the TxQ.

T4 Queue Buffer Descriptor Checksum Error

This bit is used to indicate the BD checksum error in the TxQ.

1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T3 Queue Buffer Descriptor Checksum Error

This bit is used to indicate the BD checksum error in the TxQ.

1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T2 Queue Buffer Descriptor Checksum Error

This bit is used to indicate the BD checksum error in the TxQ.

1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T1 Queue Buffer Descriptor Checksum Error

This bit is used to indicate the BD checksum error in the TxQ.

1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

T0 Queue Buffer Descriptor Checksum Error

This bit is used to indicate the BD checksum error in the TxQ.

1'b0: No BD checksum error occurred in the TxQ.
1'b1: BD checksum error occurred in the TxQ.

12.5.38 TX Queue Error Indication Mask Register (TQEIMR)

00000CB4 TQEIMR TX Queue Error Indication Mask Register 00FFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									T7Q_LENERR_IMR	T6Q_LENERR_IMR	T5Q_LENERR_IMR	T4Q_LENERR_IMR	T3Q_LENERR_IMR	T2Q_LENERR_IMR	T1Q_LENERR_IMR	T0Q_LENERR_IMR
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T7Q_GPD_CSERR_IMR	T6Q_GPD_CSERR_IMR	T5Q_GPD_CSERR_IMR	T4Q_GPD_CSERR_IMR	T3Q_GPD_CSERR_IMR	T2Q_GPD_CSERR_IMR	T1Q_GPD_CSERR_IMR	T0Q_GPD_CSERR_IMR	T7Q_BD_CSERR_IMR	T6Q_BD_CSERR_IMR	T5Q_BD_CSERR_IMR	T4Q_BD_CSERR_IMR	T3Q_BD_CSERR_IMR	T2Q_BD_CSERR_IMR	T1Q_BD_CSERR_IMR	T0Q_BD_CSERR_IMR
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
23	T7Q_LENERR_IMR	T7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
22	T6Q_LENERR_IMR	T6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
21	T5Q_LENERR_IMR	T5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
20	T4Q_LENERR_IMR	T4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
19	T3Q_LENERR_IMR	T3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
18	T2Q_LENERR_IMR	T2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.

17	T1Q_LENERR_IMR	T1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
16	T0Q_LENERR_IMR	T0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
15	T7Q_GPD_CSERR_IMR	T7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
14	T6Q_GPD_CSERR_IMR	T6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
13	T5Q_GPD_CSERR_IMR	T5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
12	T4Q_GPD_CSERR_IMR	T4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
11	T3Q_GPD_CSERR_IMR	T3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
10	T2Q_GPD_CSERR_IMR	T2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
9	T1Q_GPD_CSERR_IMR	T1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
8	T0Q_GPD_CSERR_IMR	T0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
7	T7Q_BD_CSERR_IMR	T7 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be

		unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
6	T6Q_BD_CSERR_IMR	T6 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
5	T5Q_BD_CSERR_IMR	T5 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
4	T4Q_BD_CSERR_IMR	T4 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
3	T3Q_BD_CSERR_IMR	T3 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
2	T2Q_BD_CSERR_IMR	T2 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
1	T1Q_BD_CSERR_IMR	T1 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.
0	T0Q_BD_CSERR_IMR	T0 Queue Error Indication Mask Register Writing 1 to this register, the corresponding TX queue error indication generated from QMU should be masked. 1'b0: The corresponding command queue error indication from QMU should be unmasked. 1'b1: The corresponding command queue error indication from QMU should be masked.

12.5.39 TX Queue Error Indication Mask Clear Register (TQEIMCR)

00000CB8 TQEIMCR TX Queue Error Indication Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									T7Q_LENER_IMCR	T6Q_LENER_IMCR	T5Q_LENER_IMCR	T4Q_LENER_IMCR	T3Q_LENER_IMCR	T2Q_LENER_IMCR	T1Q_LENER_IMCR	T0Q_LENER_IMCR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T7Q_GPD_	T6Q_GPD_	T5Q_GPD_	T4Q_GPD_	T3Q_GPD_	T2Q_GPD_	T1Q_GPD_	T0Q_GPD_	T7Q_BD_CS	T6Q_BD_CS	T5Q_BD_CS	T4Q_BD_CS	T3Q_BD_CS	T2Q_BD_CS	T1Q_BD_CS	T0Q_BD_C

	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	CSERR_IMCR	ERR_IMCR	ERR_IMCR	ERR_IMCR	ERR_IMCR	ERR_IMCR	ERR_IMCR	ERR_IMCR	SERR_IMCR
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23	T7Q_LENERR_IMCR	T7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
22	T6Q_LENERR_IMCR	T6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
21	T5Q_LENERR_IMCR	T5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
20	T4Q_LENERR_IMCR	T4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
19	T3Q_LENERR_IMCR	T3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
18	T2Q_LENERR_IMCR	T2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
17	T1Q_LENERR_IMCR	T1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
16	T0Q_LENERR_IMCR	T0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
15	T7Q_GPD_CSERR_IMCR	T7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
14	T6Q_GPD_CSERR_IMCR	T6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the

		Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
13	T5Q_GPD_CSERR_IMCR	T5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
12	T4Q_GPD_CSERR_IMCR	T4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
11	T3Q_GPD_CSERR_IMCR	T3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
10	T2Q_GPD_CSERR_IMCR	T2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
9	T1Q_GPD_CSERR_IMCR	T1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
8	T0Q_GPD_CSERR_IMCR	T0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
7	T7Q_BD_CSERR_IMCR	T7 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
6	T6Q_BD_CSERR_IMCR	T6 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
5	T5Q_BD_CSERR_IMCR	T5 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
4	T4Q_BD_CSERR_IMCR	T4 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

3	T3Q_BD_CSERR_IMCR	T3 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
2	T2Q_BD_CSERR_IMCR	T2 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
1	T1Q_BD_CSERR_IMCR	T1 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
0	T0Q_BD_CSERR_IMCR	T0 Queue Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

12.5.40 TX Queue Error Indication Mask Set Register (TQEIMSR)

00000CBC TQEIMSR TX Queue Error Indication Mask Set Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									T7Q_LENERR_IMSR	T6Q_LENERR_IMSR	T5Q_LENERR_IMSR	T4Q_LENERR_IMSR	T3Q_LENERR_IMSR	T2Q_LENERR_IMSR	T1Q_LENERR_IMSR	T0Q_LENERR_IMSR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	T7Q_GPD_CSERR_IMSR	T6Q_GPD_CSERR_IMSR	T5Q_GPD_CSERR_IMSR	T4Q_GPD_CSERR_IMSR	T3Q_GPD_CSERR_IMSR	T2Q_GPD_CSERR_IMSR	T1Q_GPD_CSERR_IMSR	T0Q_GPD_CSERR_IMSR	T7Q_BD_CSERR_IMSR	T6Q_BD_CSERR_IMSR	T5Q_BD_CSERR_IMSR	T4Q_BD_CSERR_IMSR	T3Q_BD_CSERR_IMSR	T2Q_BD_CSERR_IMSR	T1Q_BD_CSERR_IMSR	T0Q_BD_CSERR_IMSR
Type	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0	A0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23	T7Q_LENERR_IMSR	T7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
22	T6Q_LENERR_IMSR	T6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

21	T5Q_LENERR_IMSR	T5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
20	T4Q_LENERR_IMSR	T4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
19	T3Q_LENERR_IMSR	T3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
18	T2Q_LENERR_IMSR	T2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
17	T1Q_LENERR_IMSR	T1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
16	T0Q_LENERR_IMSR	T0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
15	T7Q_GPD_CSERR_IMSR	T7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
14	T6Q_GPD_CSERR_IMSR	T6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
13	T5Q_GPD_CSERR_IMSR	T5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
12	T4Q_GPD_CSERR_IMSR	T4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
11	T3Q_GPD_CSERR_IMSR	T3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.

		1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
10	T2Q_GPD_CSERR_IMSR	T2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
9	T1Q_GPD_CSERR_IMSR	T1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
8	T0Q_GPD_CSERR_IMSR	T0 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
7	T7Q_BD_CSERR_IMSR	T7 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
6	T6Q_BD_CSERR_IMSR	T6 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
5	T5Q_BD_CSERR_IMSR	T5 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
4	T4Q_BD_CSERR_IMSR	T4 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
3	T3Q_BD_CSERR_IMSR	T3 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
2	T2Q_BD_CSERR_IMSR	T2 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
1	T1Q_BD_CSERR_IMSR	T1 Queue Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
0	T0Q_BD_CSERR_IMSR	T0 Queue Error Indication Mask Set Register

Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register.
1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.
1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

12.5.41 TX Endpoint Error Indication Register (TEPEIR)

00000CC0 TEPEIR TX Endpoint Error Indication Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7EP_ERR	T6EP_ERR	T5EP_ERR	T4EP_ERR	T3EP_ERR	T2EP_ERR	T1EP_ERR	T0EP_ERR
Type									W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	T7EP_ERR	T7 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
6	T6EP_ERR	T6 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
5	T5EP_ERR	T5 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
4	T4EP_ERR	T4 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
3	T3EP_ERR	T3 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
2	T2EP_ERR	T2 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
1	T1EP_ERR	T1 Endpoint Error The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue. 1'b0: No error. 1'b1: Error.
0	T0EP_ERR	T0 Endpoint Error

The corresponding endpoint has an USB MAC related error which may require SW to stop the corresponding queue.
1'b0: No error.
1'b1: Error.

12.5.42 TX Endpoint Error Indication Mask Register (TEPEIMR)

00000CC4 TEPEIMR TX Endpoint Error Indication Mask Register 000000FF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7EP_ERR_IMR	T6EP_ERR_IMR	T5EP_ERR_IMR	T4EP_ERR_IMR	T3EP_ERR_IMR	T2EP_ERR_IMR	T1EP_ERR_IMR	T0EP_ERR_IMR
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									1	1	1	1	1	1	1	1

Bit(s)	Name	Description
7	T7EP_ERR_IMR	T7 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
6	T6EP_ERR_IMR	T6 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
5	T5EP_ERR_IMR	T5 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
4	T4EP_ERR_IMR	T4 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
3	T3EP_ERR_IMR	T3 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
2	T2EP_ERR_IMR	T2 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from

		USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
1	T1EP_ERR_IMR	T1 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.
0	T0EP_ERR_IMR	T0 Endpoint Error Indication Mask Register Writing 1 to this register, the corresponding TX endpoint error indication generated from USB MAC should be masked. 1'b0: The corresponding command endpoint error indication from EPMU should be unmasked. 1'b1: The corresponding command endpoint error indication from EPMU should be masked.

12.5.43 TX Endpoint Error Indication Mask Clear Register (TEPEIMCR)

00000CC8 TEPEIMCR TX Endpoint Error Indication Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7EP_ERR_IMCR	T6EP_ERR_IMCR	T5EP_ERR_IMCR	T4EP_ERR_IMCR	T3EP_ERR_IMCR	T2EP_ERR_IMCR	T1EP_ERR_IMCR	T0EP_ERR_IMCR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	T7EP_ERR_IMCR	T7 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
6	T6EP_ERR_IMCR	T6 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
5	T5EP_ERR_IMCR	T5 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
4	T4EP_ERR_IMCR	T4 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register.

		1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
3	T3EP_ERR_IMCR	T3 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
2	T2EP_ERR_IMCR	T2 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
1	T1EP_ERR_IMCR	T1 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.
0	T0EP_ERR_IMCR	T0 Endpoint Error Indication Mask Clear Register Each bit in this register is used to clear the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Clear the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is enabled thereafter.

12.5.44 TX Endpoint Error Indication Mask Set Register (TEPEIMSR)

00000CCC TEPEIMSR TX Endpoint Error Indication Mask Set Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									T7EP_ERR_IMSR	T6EP_ERR_IMSR	T5EP_ERR_IMSR	T4EP_ERR_IMSR	T3EP_ERR_IMSR	T2EP_ERR_IMSR	T1EP_ERR_IMSR	T0EP_ERR_IMSR
Type									A0	A0	A0	A0	A0	A0	A0	A0
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7	T7EP_ERR_IMSR	T7 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
6	T6EP_ERR_IMSR	T6 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
5	T5EP_ERR_IMSR	T5 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the

		Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
4	T4EP_ERR_IMSR	T4 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
3	T3EP_ERR_IMSR	T3 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
2	T2EP_ERR_IMSR	T2 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
1	T1EP_ERR_IMSR	T1 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.
0	T0EP_ERR_IMSR	T0 Endpoint Error Indication Mask Set Register Each bit in this register is used to set the corresponding indication mask bit in the Indication Mask Register. 1'b0: No effect on the corresponding indication mask bit in the Indication Mask Register. 1'b1: Set the corresponding indication mask bit in the Indication Mask Register. The corresponding indication is disabled thereafter.

12.5.45 Device Software Interrupt Mask Register (DSIMR)

00000C14 DSIMR Device Software Interrupt Mask Register 0000FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DSIMR															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
15:0	DSIMR	Device Software Interrupt Mask Register Writing 1 to this register, the corresponding software interrupt request from host should be masked. yes 0: The corresponding software interrupt request from host should be unmasked. 1: The corresponding software interrupt request from host should be masked.

12.5.46 Device Software Interrupt Mask Register (DSIMR)

00000C14 DSIMR Device Software Interrupt Mask Register 0000FFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DSIMR															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
15:0	DSIMR	Device Software Interrupt Mask Register Writing 1 to this register, the corresponding software interrupt request from host should be masked. yes 0: The corresponding software interrupt request from host should be unmasked. 1: The corresponding software interrupt request from host should be masked.

12.5.47 Device Software Interrupt Mask Register (DSIMCR)

00000C18 DSIMCR Device Software Interrupt Mask Clear Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DSIMCR															
Type	A0															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	DSIMCR	Device Software Interrupt Mask Clear Register Each bit in this register is used to clear the corresponding interrupt mask bit in the Interrupt Mask Register. 0: No effect on the corresponding interrupt mask bit in the Interrupt Mask Register. 1: Clear the corresponding interrupt mask bit in the Interrupt Mask Register. The corresponding interrupt is enabled thereafter.

12.5.48 Device Software Interrupt Mask Register (DSIMSR)

00000C1C DSIMSR Device Software Interrupt Mask Set Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	DSIMSR															
Type	A0															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	DSIMSR	Device Sofware Interrupt Mask Set Register Each bit in this register is used to set the corresponding interrupt mask bit in the Interrupt Mask Register. Note, does not set this register, 0: No effect on the corresponding interrupt mask bit in the Interrupt Mask Register. 1: Set the corresponding interrupt mask bit in the Interrupt Mask Register. The corresponding interrupt is disabled thereafter.

13 MSDC Controller

13.1 Introduction

The MSDC (SD card Controller) fully supports

- SD memory card specification version 3.0
- SDIO card specification version 3.0
- MMC/eMMC 4.5

13.2 Feature list

Each MSDC contains:

- Interface with MCU by AHB bus
- 32-bit access on AHB bus
- 32-bit access for control registers
- 8-bit/16-bit/32-bit access for FIFO in PIO mode
- Built-in 128 bytes FIFO buffers for transmit and receive
- Built-in CRC circuit
- Basic DMA mode, basic descriptor mode, and enhanced descriptor mode for SD/MMC
- Interrupt capabilities
- Does not support SPI mode for SD/MMC memory card
- Does not support suspend/resume for SD/MMC memory card
- Supports SD3.0 SDR104, data rate up to 208x4Mbps
- Support sSD3.0 DDR50, data rate up to 50x4x2Mbps(4-bit with clock dual edge)
- Supports e-MMC boot-up mode
- 256 programmable serial clock rates on SD/MMC bus from 100kHz to 208MHz

The configurations of 2x MSDC ports are below:

Table 13-1: MSDC Configuration Table

MSDC0	MSDC1
1/2/4/8 bits	1/2/4 bits
eMMC4.5	SD 3.0
	SDIO 3.0

13.3 Block Diagram

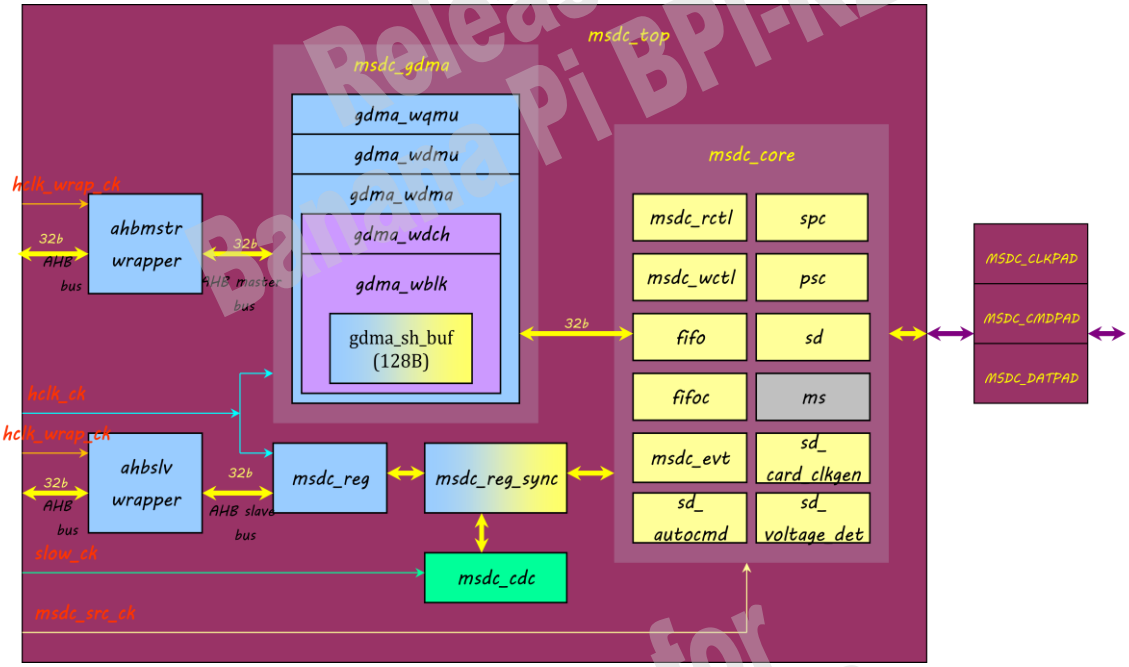


Figure 13-1: Block Diagram of MSDC controller

13.4 Register Definition

Table 13-2: MSDC Base Address Table

MSDC number	Base address	Feature
MSDC0	0x11230000	EMMC4.5
MSDC1	0x11240000	SD3.0/SDIO3.0

There are 3 MSDC IPs in this SOC. Use of the registers below are the same except that the base address needs to be changed to respective one.

Module name: MSDC Base address: (+11230h)

Address	Name	Width	Register Function
11230000	**MSDC_CFG**	32	**MSDC Configuration Register** The register is for general configuration of the MS/SD controller.
11230004	**MSDC_IOCON**	32	**MSDC IO Configuration Register** The register contains the receiver path data latch timing control and interface control bits.
11230008	**MSDC_PS**	32	**MSDC Pin Status Register** The register is used to storing card detection and write protection pin status. Card detection status can be disabled.
1123000C	**MSDC_INT**	32	**MSDC Interrupt Register** The register contains the status of interrupts. Note that the register still shows the status of interrupt even though the interrupt is disabled.

11230010	**MSDC_INTEN**	32	**MSDC Interrupt Enable Register** The register contains the related enable bit of interrupts.
11230014	**MSDC_FIFOCS**	32	**MSDC FIFO Control and Status Register** The register contains the control and status of embedded 128B FIFO.
11230018	**MSDC_TXDATA**	32	**MSDC TX Data Port Register** The register is for PIO mode only. Used to input MSDC write data to card. The access can be AHB 1B/2B/4B
1123001C	**MSDC_RXDATA**	32	**MSDC RX Data Port Register** The register is for PIO mode only. Used to read back MSDC read data from card. The access can be AHB 1B/2B/4B.
11230030	**SDC_CFG**	32	**SD Configuration Register** The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
11230034	**SDC_CMD**	32	**SD Command Register** The register defines a SD Memory Card command and its attributes. Before MS/SD controller issues a transaction onto SD bus, application shall specify other relative settings such as argument for command. After writing the register by the application, MS/SD controller will issue the corresponding transaction onto SD serial bus. If the command is GO_IDLE_STATE, the controller will have serial clock on SD/MMC bus run 128 cycles before issuing the command.
11230038	**SDC_ARG**	32	**SD Argument Register** The register contains the argument of the SD/MMC Memory Card command.
1123003C	**SDC_STS**	32	**SD Status Register** The register reflects SD bus status and contains MMC stream write status.
11230040	**SDC_RESP0**	32	**SD Response Register 0** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11230044	**SDC_RESP1**	32	**SD Response Register 1** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11230048	**SDC_RESP2**	32	**SD Response Register 2** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
1123004C	**SDC_RESP3**	32	**SD Response Register 3** The register contains parts of the last SD/MMC Memory Card bus response. The register fields SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3 are composed of the last SD/MMC Memory card bus response. For response of type R2, that is, response of the command ALL_SEND_CID, SEND_CSD and SEND_CID, only bit 127 to 0 of response token is stored in the register field SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3. SDC_RESP0 = bit 31~0 SDC_RESP1 = bit 63~32 SDC_RESP2 = bit 95~64 SDC_RESP3 = bit 127~96 For response of type R1b in auto CMD12 or R1 in auto CMD23, bit 39 to 8 of response token is stored in the register field of SDC_RESP3. For the responses of other types, only bit 39 to 8 of response token is stored in the register field SDC_RESP0.
11230050	**SDC_BLK_NUM**	32	**SD Block Number Register**

			This register defines the block number for the block transaction. For single read/write, this register should be set to 1. For multiple read/write, this register should be set to larger than 1. Set to 0 will cause unexpected result.
11230054	SDC_VOL_CHG	32	**SD Voltage Change Wait Time Register** This register define SD voltage change check wait time
11230058	SDC_CSTS	32	**SD Card Status Register** After commands with R1 and R1b response, this register will contain the status of the SD/MMC card
1123005C	SDC_CSTS_EN	32	**SD Card Status Enable Register** This register is used to control which bit of the SDC_CSTS will generate the MSDC_INT.SD_CSTA interrupt.
11230060	SDC_DATCRC_STS	32	**SD Card Data CRC Status Register** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read
11230070	EMMC_CFG0	32	**EMMC Configuration Register 0** The register is used for boot up mode general configuration of e-MMC version 4.3 and 4.4.
11230074	EMMC_CFG1	32	**EMMC Configuration Register 1** The register is used for boot up mode general configuration of e-MMC version 4.3 and 4.4.
11230078	EMMC_STS	32	**EMMC Status Register** The register reflects the status of e-MMC boot up mode operation.
1123007C	EMMC_IOCON	32	**EMMC IO Control Register** The register controls the H/W reset pin of e-MMC boot up mode operation.
11230080	SD_ACMD_RESP	32	**SD ACMD Response Register** This register stores the response of auto command from SD card
11230084	SD_ACMD19_TRG	32	**SD ACMD19 Target Register** This register is used to select target delay line to run ACMD19 sequence.
11230088	SD_ACMD19_STS	32	**SD ACMD19 Status Register** This register stores the result of auto command 19 from SD card
1123008C	DMA_SA_HIGH4BIT	32	**DMA Current Address Resgiter of high 4bit** This register contain the start address high 4bit of 36bit address for 64G dram access
11230090	DMA_SA	32	**DMA Start Address Register** This register contains the start address of the DMA descriptor
11230094	DMA_CA	32	**DMA Current Address Register** This register contains the current DMA address
11230098	DMA_CTRL	32	**DMA Control Register** This register is used to control the DMA operation.
1123009C	DMA_CFG	32	**DMA Configuration Register** This register is used to configure the DMA operation.
112300A0	SW_DBG_SEL	32	**MSDC S/W Debug Selection Register** This register is used to select S/W debug output
112300A4	SW_DBG_OUT	32	**MSDC S/W Debug Output Register** This register shows the selected debug output
112300A8	DMA_LENGTH	32	**DMA Length Register** This register is used to set Basic DMA operation length
112300B0	PATCH_BIT0	32	**MSDC Patch Bit Register 0** This register can configure the patch function. For normal function, these bit should keep in default value
112300B4	PATCH_BIT1	32	**MSDC Patch Bit Register 1**

			This register can configure the patch function. For normal function, these bit should keep in default value
112300B8	PATCH_BIT2	32	**MSDC Patch Bit Register 2** This register can configure the patch function. For normal function, these bit should keep in default value
112300C0	DAT0_TUNE_CRC	32	**DAT0 Tune Result Register** This register record on-line tuning result for DAT0 line
112300C4	DAT1_TUNE_CRC	32	**DAT1 Tune Result Register** This register record on-line tuning result for DAT1 line
112300C8	DAT2_TUNE_CRC	32	**DAT2 Tune Result Register** This register record on-line tuning result for DAT2 line
112300CC	DAT3_TUNE_CRC	32	**DAT3 Tune Result Register** This register record on-line tuning result for DAT3 line
112300D0	CMD_TUNE_CRC	32	**CMD Tune Result Register** This register record on-line tuning result for CMD line
112300D4	SDIO_TUNE_WIND	32	**SDIO Tune Window Register 0** This register define tuning window size for SDIO on-line CRC tuning feature
112300F0	PAD_TUNE0	32	**MSDC Pad Tuning Register0** This register can configure the delay line embedded in Pad Macro
112300F4	PAD_TUNE1	32	**MSDC Pad Tuning Register1** This register can configure the delay line embedded in Pad Macro
112300F8	DAT_RD_DLY0	32	**MSDC Data Delay Line Register 0** This register can configure the delay line embedded in Pad Macro
112300FC	DAT_RD_DLY1	32	**MSDC Data Delay Line Register 1** This register can configure the delay line embedded in Pad Macro
11230100	DAT_RD_DLY2	32	**MSDC Data Delay Line Register 2** This register can configure the delay line embedded in Pad Macro
11230104	DAT_RD_DLY3	32	**MSDC Data Delay Line Register 3** This register can configure the delay line embedded in Pad Macro
11230110	HW_DBG_SEL	32	**MSDC H/W Debug Selection Register** This register can select the H/W debug output
11230114	MAIN_VER	32	**MSDC Main Version Register** This register shows the version code of MSDC IP
11230118	ECO_VER	32	**MSDC ECO Version Register** This register shows the ECO version code of MSDC IP

11230000 MSDC_CFG MSDC Configuration Register 02000099

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							SCLKSTOPSEL				CCKMD		CCKDIV[11:8]			
Type							RW				W1C		RW			
Reset							1				0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCKDIV[7:0]								CCKSB	BV18PSS	BV18SDT	CCKDRVE	PIO	RST	CCKPD	MSDC
Type	RW								RU	RU	RW	RW	RW	A0	RW	RW
Reset	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1

Bit(s)	Mnemonic	Name	Description
25	SCLKSTOP	SCLK_STOP_SEL	In DDR mode,stop SCLK when device is idle whether check

Bit(s)	Mnemonic	Name	Description
	SEL		**SCLK phase** 1'b0: stop sclk no check SCLK phase 1'b1: stop sclk check SCLK phase,fix 1/4T sclk glitch in DDR mode
21:20	**CCKMD**	CARD_CK_MODE	**MS/SD Card clock mode** 2'b00: Use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b01: Use msdc_src_ck as msdc_ck, bit[15]~bit[8] is ignored. 2'b10: DDR mode, also use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b11: HS400 mode,also use clock divider output and use msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed.
19:8	**CCKDIV**	CARD_CK_DIV	**MS/SD Card clock divider** The register field controls clock frequency of serial clock on MS/SD bus. Please refer to Data Line Latching Timing Diagram and Response Latching Timing Diagram. For non-DDR mode, msdc_ck equals SD bus clock. (Ex: For SDR25 or HS, msdc_ck and SD bus clock will be 50MHz) For DDR mode, msdc_ck denotes the MSDC internal clock which will be dobule to SD bus clock. (Ex: For DDR50, msdc_ck should be set to 100MHz and bus clock will be 50MHz) 8'b00000000: msdc_ck =(1/2) * msdc_src_ck 8'b00000001: msdc_ck = (1/(4*1)) * msdc_src_ck 8'b00000010: msdc_ck = (1/(4*2)) * msdc_src_ck 8'b00000011: msdc_ck = (1/(4*3))* msdc_src_ck 8'b00010000: msdc_ck = (1/(4*16))* msdc_src_ck 8'b11111111: msdc_ck = (1/(4*255)) * msdc_src_ck
7	**CCKSB**	CARD_CK_STABLE	**MS/SD Card clock stable or not** After programming the CARD_CK_MODE or CARD_CK_DIV, this bit will immediately go to "0" and return to "1" if stable. User should poll this register to make sure the safety control of MSDC. 1'b0: Clock output is not stable 1'b1: Clock output is stable
6	**BV18PSS**	BV_1P8_PASS	**MSDC Bus voltage 1.8V detection status** S/W should check this bit after BUS_VOL_18V_START_DET turns to 0 from 1. 1'b0: The voltage detection has error. 1'b1: The voltage detection has no error.
5	**BV18SDT**	BV_1P8_START_DET	**MSDC Bus voltage 1.8V detection sequence start event** S/W writes this bit to 1 to trigger H/W outputs 1.8V clock for 1 ms and automatically detect CMD/DAT line sequence for voltage change is passed or not. H/Q will clear this bit to 0 after the detection has finished. The pass or fail status is stored in bit[6] BUS_VOL_18V_PASS.
4	**CCKDRVE**	CARD_CK_DRV_EN	**SD/MS Card Bus Clock drive enable bit** Set this bit to 1 to enable MSDC bus clock driver. The default bus state depends on MSDC_CFG][1] CARD_CK_PWDN bit. If MSDC_CFG[1] CARD_CK_PWDN= 1, the default clock state is free running. If MSDC_CFG[1] CARD_CK_PWDN = 0, the default clock state is gated to 0. Set this bit to 0 will put the bus state into "tri-state". Default is 1. 1'b0: Put the clock pad into tri-state 1'b1: Enable MSDC to drive clock pad, the state of CLK depends on MSDC_CFG[1] CARD_CK_PWDN
3	**PIO**	PIO_MODE	**MS/SD PIO mode** PIO mode selection. Default is in PIO mode. 1'b0: DMA mode

Bit(s)	Mnemonic	Name	Description
			1'b1: PIO mode
2	**RST**	RST	**Software reset** Writing 1 to this register will cause internal synchronous reset of MS/SD controller, and it will not reset register settings and DMA controller. The reset sequence is done when this bit goes to 0. S/W should wait this bit back to 0 after writing 1. 1'b0: MS/SD controller is not in reset state 1'b1: MS/SD controller is in reset state
1	**CCKPD**	CARD_CK_PWDN	**MSDC bus clock power down mode** This bit controls the card clock power down mode. 1'b0: Clock is gated to 0 if no command or data is transmitted. 1'b1: Clock is free running even if no command or data is transmitted. (The clock may still be stopped when MSDC write data is not enough or no space for next read data)
0	**MSDC**	MSDC	**MS/SD mode selection** The register bit is used to configure the controller as the host of Memory Stick or as the host of SD/MMC Memory card. The default value is to configure the controller as the host of Memory Stick. 1'b0: Configure the controller as the host of Memory Stick 1'b1: Configure the controller as the host of SD/MMC Memory card

11230004 MSDC_IOCON MSDC IO Configuration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									RD7SPL	RD6SPL	RD5SPL	RD4SPL	RD3SPL	RD2SPL	RD1SPL	RD0SPL
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			WD3SPL	WD2SPL	WD1SPL	WD0SPL	WDSPLSEL	WDSPL			RDSPLSEL	DDR50CKD	DDLSEL	RDSPL	RSPL	SDR104CKS
Type			RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23	**RD7SPL**	R_D7_SMPL	**Read data 7 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
22	**RD6SPL**	R_D6_SMPL	**Read data 6 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
21	**RD5SPL**	R_D5_SMPL	**Read data 5 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
20	**RD4SPL**	R_D4_SMPL	**Read data 4 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
19	**RD3SPL**	R_D3_SMPL	**Read data 3 sample selection**

Bit(s)	Mnemonic	Name	Description
			This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
18	**RD2SPL**	R_D2_SMPL	**Read data 2 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
17	**RD1SPL**	R_D1_SMPL	**Read data 1 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
16	**RD0SPL**	R_D0_SMPL	**Read data 0 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
13	**WD3SPL**	W_D3_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
12	**WD2SPL**	W_D2_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
11	**WD1SPL**	W_D1_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
10	**WD0SPL**	W_D0_SMPL	**CRC Status and SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
9	**WDSPLSEL**	W_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in write transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.W_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.W_D0_SMPL Data line 1: MSDC_IOCON.W_D1_SMPL Data line 2: MSDC_IOCON.W_D2_SMPL Data line 3: MSDC_IOCON.W_D3_SMPL
8	**WDSPL**	W_D_SMPL	**CRC Status and SDIO interrupt sample selection** 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
5	**RDSPLSEL**	R_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in read transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.R_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.R_D0_SMPL Data line 1: MSDC_IOCON.R_D1_SMPL Data line 2: MSDC_IOCON.R_D2_SMPL Data line 3: MSDC_IOCON.R_D3_SMPL

Bit(s)	Mnemonic	Name	Description
			Data line 4: MSDC_IOCON.R_D4_SMPL Data line 5: MSDC_IOCON.R_D5_SMPL Data line 6: MSDC_IOCON.R_D6_SMPL Data line 7: MSDC_IOCON.R_D7_SMPL
4	**DDR50CKD**	DDR50_DLY_SEL	**DDR50 output clock delay selection** 1'b0: Use default clock output 1'b1: Delay 1T msdc_src_ck for clock output
3	**DDLSEL**	D_DLYLINE_SEL	**Data line delay line fine tune selection** 1'b0: All data line share one delay selection value indicated by PAD_TUNE.PAD_DAT_RD_RXDLY 1'b1: Each data line has its own delay selection value indicated by Data line 0: DAT_RD_DLY0.DAT0_RD_DLY Data line 1: DAT_RD_DLY0.DAT1_RD_DLY Data line 2: DAT_RD_DLY0.DAT2_RD_DLY Data line 3: DAT_RD_DLY0.DAT3_RD_DLY Data line 4: DAT_RD_DLY1.DAT4_RD_DLY Data line 5: DAT_RD_DLY1.DAT5_RD_DLY Data line 6: DAT_RD_DLY1.DAT6_RD_DLY Data line 7: DAT_RD_DLY1.DAT7_RD_DLY
2	**RDSPL**	R_D_SMPL	**Read data sample selection** 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
1	**RSPL**	R_SMPL	**Command response sample selection** 1'b0: Sample response by external bus clock rising edge 1'b1: Sample response by external bus clock falling edge
0	**SDR104CKS**	SDR104_CLK_SEL	**SDR104 SCLK output clock control** This bit is only used when MSDC_CFG[17:16] CARD_CK_MODE is 2'b01. 1'b0: Bus clock output equals inverted msdc_src_ck 1'b1: Bus clock output equals msdc_src_ck

11230008 MSDC_PS MSDC Pin Status Register 01FF0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SDWP							CMD	DAT							
Type	RU							RU	RU							
Reset	0							1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CDDBCE														CDSTS	CDEN
Type	RW														RU	RW
Reset	0	0	0	0											1	0

Bit(s)	Mnemonic	Name	Description
31	**SDWP**	SD_WP	**Write Protection Switch status on SD Memory Card** The register bit shows the status of Write Protection Switch on SD Memory Card. There is no default reset value. The pin WP (Write Protection) is only useful while the controller is configured for SD Memory Card 1'b0: Write Protection Switch ON. It means that memory card is desired to be write-protected 1'b1: Write Protection Switch OFF. It means that memory card is writable
24	**CMD**	CMD	**Command line status** This bit reflects the command line value of MSDC bus.

Bit(s)	Mnemonic	Name	Description
23:16	**DAT**	DAT	**Data line status** This bit reflects the data line value of MSDC bus. (8-bits)
15:12	**CDDBCE**	CDDEBOUNCE	**Card detection de-bounce timer** The register field specifies the time interval for card detection de-bounce. Its default value is 0. It means that de-bounce interval is one 32KHz cycle. The interval will extend one cycle time of 32KHz by increasing the counter by 1
1	**CDSTS**	CDSTS	**Card detection status** 1'b1: Card detection pin status is logic high 1'b1: Card detection pin status is logic high
0	**CDEN**	CDEN	**Card detect enable** The register bit is used to control the card detection circuit 1'b0: Card detection is disable 1'b1: Card detection is enable

1123000C MSDC_INT MSDC Interrupt Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										AC53FAIL	AC53DONE	GEAROUTBOUND	DMAPROTECT	GPDCSERR	BDCSERR	AC19DONE
Type										W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SDDCRCERR	SDDTO	DMAXFDNE	SDXFCPL	SDCSTS	SDRCRCER	SDCTO	SDCRDY	SDIOIRQ	DMAQEPTY	SDACDRCRCER	SDACDCTO	SDACDCRDY		MSDCCDSC	MMCIRQ
Type	W1C	W1C	W1C	W1C	RU	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C		W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**AC53FAIL**	AUTOCMD53_FAIL	**SD Auto command 53 status register** When auto-command 53 tuning sequence is failure , this register will be set to 1 by H/W.
21	**AC53DONE**	AUTOCMD53_DONE	**SD Auto command 53 status register** When auto-command 53 tuning sequence is finished , this register will be set to 1 by H/W.
20	**GEAROUTBOUND**	GEAR_OUT_BOUND	**the gear setting of delayline is out of boundary during SDIO Autocmd 53 On-Line tuning process**
19	**DMAPROTECT**	DMA_PROTECT	**there is write operation to DMA start address, length, start bit or last buf bit**
18	**GPDCSERR**	GPD_CS_ERR	**GPD checksum error detected**
17	**BDCSERR**	BD_CS_ERR	**BD checksum error detected**
16	**AC19DONE**	AUTOCMD19_DONE	**SD Auto command 19 status register** When auto-command 19 is enabled, H/W will clear this register to 0. As the tuning sequence is finished (32 times), this register will be set to 1 by H/W. S/W should check the AUTOCMD_STS0 and SUTOCMD_STS1 only when this bit is ON.
15	**SDDCRCERR**	SD_DATA_CRCERR	**SD Data CRC error interrupt** Indicates that MS/SD controller detects a CRC error after reading a block of data from the DAT line or SD/MMC signals a CRC error after writing a block of data to the DAT line.

Bit(s)	Mnemonic	Name	Description
			1'b0: Otherwise 1'b1: MS/SD controller detected a CRC error after reading a block of data from the DAT line or SD/MMC signaled a CRC error after writing a block of data to the DAT line
14	**SDDTO**	SD_DATTO	**SD Data timeout interrupt** Indicates that SD/MMC controller detects a timeout condition while waiting for data token on the DAT line. This bit is for both data read and data write. For SD data read, timeout will occur when the read data is not presented. For SD data write, timeout will occur when the write data CRC status is not presented if PATCH_BIT[30] DETECT_WR_CRC_TIMEOUT = 1 1'b0: Otherwise 1'b1: SD/MMC controller detects a timeout condition while waiting for data token on the DAT line
13	**DMAXFDNE**	DMA_XFER_DONE	**DMA transfer done interrupt** The register bit indicates the status of data block transfer. 1'b0: Otherwise 1'b1: A data block was successfully transferred
12	**SDXFCPL**	SD_XFER_COMPLETE	**SD Data transfer complete interrupt** This bit indicates the transaction which contains data has completed. While performing tuning procedure (Execute Tuning is set to 1), SD_XFER_COMPLETE is not set to 1.
11	**SDCSTS**	SD_CSTS	**SD CSTA update interrupt** The register bit indicates any bit in the register SDC_CSTA is active, the register bit will be set to 1. S/W should clear the SDC_CSTA and this bit will be de-asserted automatically. 1'b0: No SD Memory Card interrupt 1'b1: SD Memory Card interrupt exists
10	**SDRCRCER**	SD_RESP_CRCERR	**SD Command CRC error interrupt** Indicates that SD/MMC controller detected a CRC error after reading a response from the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a CRC error after reading a response from the CMD line
9	**SDCTO**	SD_CMDTO	**SD Command timeout interrupt** Indicates that SD/MMC controller detected a timeout condition while waiting for a response on the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a timeout condition while waiting for a response on the CMD line
8	**SDCRDY**	SD_CMDRDY	**SD Command ready interrupt** For the command without response, the register bit will be 1 once the command completes on SD/MMC bus. For command with response without busy, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error. For command with response with busy in DAT0, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error and the DAT0 transited from busy to idle. 1'b0: Otherwise 1'b1: Command finish successfully without a CRC error
7	**SDIOIRQ**	SD_SDIOIRQ	**SD SDIO interrupt** This bit indicates the interrupt is sensed in the SDIO bus.

Bit(s)	Mnemonic	Name	Description
			1'b0: No interrupt on SDIO bus 1'b1: Interrupt on SDIO bus
6	**DMAQEPTY**	DMA_Q_EMPTY	**DMA queue empty interrupt** This bit is used to indicate the current DMA queue is empty. Only for Descriptor mode and Enhance mode.
5	**SDACDRCRCER**	SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt** This bit is set when detecting a CRC error in the Auto command response.
4	**SDACDCTO**	SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt** This bit is set if no response is returned within a specified cycles(64T in spec) from the end bit of Auto command.
3	**SDACDCRDY**	SD_AUTOCMD_CMDRDY	**SD auto command ready interrupt** This bit is set if auto command is executed without CRC error or time out.
1	**MSDCCDSC**	MSDC_CDSC	**MSDC Card detection status change interrupt** The register bit indicates if any interrupt for memory card insertion/removal exists. Whenever memory card is inserted or removed and card detection circuit is enabled, i.e., the register bit CDEN in the register MSDC_PS is set to 1, the register bit will be set to 1. It will be reset when the register is read. 1'b0: Otherwise 1'b1: Card is inserted or removed
0	**MMCIRQ**	MMC_IRQ	**MMC card interrupt** 1'b0: Otherwise 1'b1: indicates that MMC card interrupt event occurs

11230010 MSDC_INTEN MSDC Interrupt Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										ENAC53FAIL	ENAC53DONE	ENGEAROUTBOUND	ENDMAPROTECT	ENGPDCSERR	ENBDCSERR	ENAC19DONE
Type										RW	RW	RW	RW	RW	RW	RW
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENSDDCRCERR	ENSDDTO	ENDMAXFDNE	ENSDXFCPL	ENSDCSTA	ENSDRCRCER	ENSDCTO	ENSDCRDY	ENSDIOIRQ	ENDMAQEPTY	ENSDACDRCRCER	ENSDACDCTO	ENSDACDCRDY		ENMSDCCDSC	ENMMCIRQ
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**ENAC53FAIL**	EN_AUTOCMD53_FAIL	**Auto-command 53 failure interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
21	**ENAC53DONE**	EN_AUTOCMD53_DONE	**Auto-command 53 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
20	**ENGEAROUTBOUND**	EN_GEAR_OUT_BOUND	**Gear out boundary interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

Bit(s)	Mnemonic	Name	Description
19	**ENDMAPROTECT**	EN_DMA_PROTECT	**DMA protection interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
18	**ENGPDCSERR**	EN_GPD_CS_ERR	**GPD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
17	**ENBDCSERR**	EN_BD_CS_ERR	**BD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
16	**ENAC19DONE**	EN_AUTOCMD19_DONE	**Auto-command 19 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
15	**ENSDDCRCERR**	EN_SD_DATA_CRCERR	**SD Data CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
14	**ENSDDTO**	EN_SD_DATTO	**SD Data timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
13	**ENDMAXFDNE**	EN_SD_DMA_XFER_DONE	**DMA transfer done interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
12	**ENSDXFCPL**	EN_SD_XFER_COMPLETE	**SD Data transfer complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
11	**ENSDCSTA**	EN_SD_CSTA	**SD CSTA update interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
10	**ENSDRCRCER**	EN_SD_RESP_CRCERR	**SD Command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
9	**ENSDCTO**	EN_SD_CMDTO	**SD Command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
8	**ENSDCRDY**	EN_SD_CMDRDY	**SD Command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
7	**ENSDIOIRQ**	EN_SD_SDIOIRQ	**SD SDIO interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
6	**ENDMAQEPTY**	EN_DMA_Q_EMPTY	**DMA queue empty interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
5	**ENSDACDRCRCER**	EN_SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
4	**ENSDACDCTO**	EN_SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
3	**ENSDACDCRDY**	EN_AUTOCMD_CMDRDY	**SD auto command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

Bit(s)	Mnemonic	Name	Description
1	**ENMSDCCDSC**	EN_MSDC_CDSC	**MSDC Card detection status change interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
0	**ENMMCIRQ**	EN_MMC_IRQ	**MMC card interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

11230014 MSDC_FIFOCS MSDC FIFO Control and Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFOCLR								TXFIFOCNT							
Type	A0								RU							
Reset	0								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXFIFOCNT							
Type									RU							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**FIFOCLR**	FIFOCLR	**Embedded FIFO clear** Write this bit to 1 makes FIFO cleared. It will goes to 0 when FIFO is cleared. S/W needs to check this bit to make sure clearing FIFO sequence is done. This bit can be used when the data read/write sequence has error and need to clean the H/W FIFO.
23:16	**TXFIFOCNT**	TXFIFOCNT	**TX FIFO count for MSDC write** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved
7:0	**RXFIFOCNT**	RXFIFOCNT	**RX FIFO count for MSDC read** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved

11230018 MSDC_TXDATA MSDC TX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIOTXDATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIOTXDATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
31:0	**PIOTXDATA**	PIO_TXDATA	**PIO mode TXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

1123001C MSDC_RXDATA MSDC RX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIORXDATA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIORXDATA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIORXDATA**	PIO_RXDATA	**PIO mode RXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

11230030 SDC_CFG SD Configuration Register 00100000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DTOC										INTBGP	SDIOIDE	SDIO		BUSWD	
Type	RW										RW	RW	RW		RW	
Reset	0	0	0	0	0	0	0	0			0	1	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ENWKUPINS	ENWKUPSDIOINT
Type															RW	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31:24	**DTOC**	DTOC	**Data Timeout Counter** The period from the end of the initial host read command or the last read data block in a multiple block read operation to the start bit of the next read data block requires at least two serial clock cycles. The counter is used to extend the period (Read Data Access Time) in unit of 1048576 serial clocks. 8'b00000000: Extend 1048576 more serial clock cycle 8'b00000001: Extend 1048576x2 more serial clock cycle 8'b00000010: Extend 1048576x3 more serial clock cycle 8'b11111111: Extend 1048576x 256 more serial clock cycle
21	**INTBGP**	INT_AT_BLOCK_GAP	**Interrupt at block Gap** This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. Setting to 0 disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the Host Driver detects an SD card

Bit(s)	Mnemonic	Name	Description
			insertion, it shall set this bit according to the CCCR of the SDIO card. 1'b0: Disables interrupt detection at the block gap 1'b1: Enables interrupt detection at the block gap
20	**SDIOIDE**	SDIO_INT_DET_EN	**SDIO interrupt detection enable** This bit is to inform the SD controller to sense the SDIO interrupt 1'b0: SDIO interrupt detection is disabled 1'b1: SDIO interrupt detection is enabled if the SDIO bit is also on
19	**SDIO**	SDIO	**SDIO mode enable bit** This bit is to enable the support to sense the SDIO interrupt and disable the R4 response CRC check for SDIO card 1'b0: SDIO mode is disabled 1'b1: SDIO mode is enabled
17:16	**BUSWD**	BUSWIDTH	**Bus width configuration** This field is used to define the SD/MMC bus width 2'b00: 1 bit mode 2'b01: 4 bit mode 2'b10: 8 bit mode 2'b11: reserved
1	**ENWKUPINS**	WAKEUP_INS_EN	**Card status change wakeup event enable bit** 1'b0: Disable wakeup event for card status change 1'b1: Enable wakeup event for card status change
0	**ENWKUPSDIOINT**	WAKEUP_SDIOINT_EN	**SDIO card interrupt wakeup event enable bit** 1'b0: Disable wakeup event for SDIO card interrupt 1'b1: Enable wakeup event for SDIO card interrupt

11230034 SDC_CMD SD Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD53	VOLSWTH	ACMD		LEN											
Type	RW	RW	RW		RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOIRQ	STOP	RW	DTYPE			RSPTYP			BREAK	CMD					
Type	RW	RW	RW	RW			RW			RW	RW					
Reset	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**ACMD53**	AUTO_CMD53	**Auto command 53 enable** This field should use exclusively with AUTO_CMD field. means that AUTO_CMD != 2'b00, this field should assign to 1'b0, if this filed is assigned to 1'b1, AUTO_CMD should set to 2'b00 1'b0: Disable Auto Command 53 1'b1: Enable Auto Command 53
30	**VOLSWTH**	VOL_SWTH	**Voltage switch command** 1'b0: Disable voltage switch detection 1'b1: Enable voltage switch detection
29:28	**ACMD**	AUTO_CMD	**Auto command enable** This field determines use of auto command functions. This function can be used in all modes including PIO/Basic DMA/Descriptor DMA/Enhanced Mode. There are two methods to stop Multiple-block read and write operation.

Bit(s)	Mnemonic	Name	Description
			(1) Auto CMD12 Enable Multiple-block read and write commands for memory require CMD12 to stop the operation. When ACMD-12 is used, MSDC issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the MSDC_INT register. The Host Driver shall not set this bit if the command does not require CMD12. In particular, secure commands defined in the Part 3 File Security specification do not require CMD12. (2) Auto CMD23 Enable When ACMD-23 is used, MSDC issues a CMD23 automatically before issuing a command specified in the CMD field. The Host Controller Version 3.00 and later shall support this function. By writing the Command register, MSDC issues a CMD23 first and then issues a command specified by the CMD field in SDC_CMD register. If response errors of CMD23 are detected, the second command is not issued. A CMD23 error is indicated in the MSDC_INT register. 32-bit block count value for CMD23 is set to SDC_BLOCK_NUM register. 2'b00: Disable Auto Command 2'b01: Enable Auto CMD12 2'b10: Enable Auto CMD23 2'b11: Enable Re-tuning CMD19
27:16	**LEN**	LEN	**Length** The register field is used to define the length of one block in unit of byte in a data transaction of block mode or the data length in unit of byte in data transaction of byte mode. The maximal value of block length is 2048 bytes. 12'b000000000000: Reserved 12'b000000000001: Block length is 1 byte 12'b000000000010: Block length is 2 byte 12'b011111111111: Block length is 2047 byte 12'b100000000000: Block length is 2048 byte
15	**GOIRQ**	GO_IRQ	**GO_IRQ command** The register bit indicates if the command is GO_IRQ_STATE (CMD40) and used only for MMC protocol. If the command is GO_IRQ_STATE, the period between command token and response token will not be limited. 1'b0: The command is not GO_IRQ_STATE 1'b1: The command is GO_IRQ_STATE
14	**STOP**	STOP	**Stop command** The register bit indicates if the command is a stop transmission command. It should be set to 1 when CMD12 (SD/MMC) or CMD52 with I/O abort (SDIO) is to be issued. 1'b0: The command is not a stop transmission command 1'b1: The command is a stop transmission command
13	**RW**	RW	**Command read write selection** The register bit defines the command is a read command or write command. The register bit is valid only when the command will cause a transaction with data token. 1'b0: The command is a read command 1'b1: The command is a write command
12:11	**DTYPE**	DTYPE	**Data block selection** The register field defines data token type for the command. 2'b00: No data token for the command 2'b01: Single block transaction (only available in block mode) 2'b10: Multiple block transaction. (only available in block mode) 2'b11: Stream operation. It only shall be used in MMC protocol. (only available in block mode)

Bit(s)	Mnemonic	Name	Description
9:7	**RSPTYP**	RSPTYP	**Command response type** 3'b000: This command has no response. 3'b001: The command has R1/R5/R6/R7 response. The response token is 48-bit with CRC check (For SD/MMC/SDIO) (Not include the SDIO abort command) 3'b010: The command has R2 response. The response token is 136-bit (For SD/MMC) 3'b011: The command has R3 response. The response token is 48-bit response, no CRC check (For SD/MMC) 3'b100: The command has R4 response. The response token is 48-bit without CRC check (For SDIO) The response token is 48-bit with CRC check (For MMC) 3'b111: The command has R1b response. The response token is 48-bit (For SD/MMC/SDIO)
6	**BREAK**	BREAK	**Abort a pending MMC GO_IRQ command** It is only valid for a pending GO_IRQ_MODE command waiting for MMC interrupt response. 1'b0: Not a beak command 1'b1: Break a pending MMC GO_IRQ_MODE command in the controller.
5:0	**CMD**	CMD	**SD Memory Card command**

11230038 SDC_ARG SD Argument Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARG[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ARG[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ARG**	ARG	**Memory card controller argument register**

1123003C SDC_STS SD Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMCSWRCPL															CMD_WR_BUSY
Type	RU															W1C
Reset	0															0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															CMDBSY	SDCBSY
Type															RU	RU
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31	**MMCSWRCPL**	MMC_STREAM_WR_COMPL	**MMC Stream mode write data is all flushed to MMC card** S/W can use this bit to confirm last write data are flushed to MMC then issue STOP command.

Bit(s)	Mnemonic	Name	Description
			This bit is only valid when the command SDC_CMD.DTYPE=2'b11. 1'b0: Last Data are partially inside MSDC 1'b1: Last data are flushed to MMC card
16		CMD_WR_BUSY	
1	**CMDBSY**	CMDBUSY	**SD Command line busy status** S/W should always read this bit to make sure the command line is not busy before sending the next command. If the command is R1B or data read/write command, S/W should check SDCBUSY bit too. Note: When Auto command 12 is enabled, this bit will be asserted immediately after SDC_CMD is written and de-asserted after auto-command 12 finishes. 1'b0: No transmission is going on CMD line on SD bus 1'b1: There exists transmission going on CMD line on SD bus
0	**SDCBSY**	SDCBUSY	**SD controller busy status** 1'b0: SD controller is idle 1'b1: SD controller is busy

11230040 SDC_RESP0 SD Response Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP0**	RESP0	**Memory card controller response register 0**

11230044 SDC_RESP1 SD Response Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP1[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP1[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP1**	RESP1	**Memory card controller response register 1**

11230048 SDC_RESP2 SD Response Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP2[31:16]															
Type	RU															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP2[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP2**	RESP2	**Memory card controller response register 2**

1123004C SDC_RESP3 SD Response Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP3[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP3[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP3**	RESP3	**Memory card controller response register 3**

11230050 SDC_BLK_NUM SD Block Number Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BLKNUM[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BLKNUM[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0	**BLKNUM**	BLOCK_NUMBER	**Memory card controller Block number** This field indicates the block number of data transaction. 32'd0: Reserved 32'd1: 1 data block 32'd2: 2 data block 32'd3: 3 data block 32'hffffffff: 4GB-1 data block

11230054 SDC_VOL_CHG SD Voltage Change Wait Time Register 00000145

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VCHGCNT															
Type	RW															

Reset	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15:0	**VCHGCNT**	VOL_CHG_WAIT_CNT	**This register define SD voltage change check wait time,wait time is clock frequency multiply VOL_WAIT_TIME**

11230058 SDC_CSTS SD Card Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS**	CSTS	**CSTS** The card status field in the response R1 or R1b field. Each bit can be write 1 clear individually.

1123005C SDC_CSTS_EN SD Card Status Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS_EN[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS_EN[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS_EN**	CSTS_EN	**CSTS_EN** This register is used to control which bit of the CSTA will generate the MSDC_INT.SDCSTA

11230060 SDC_DATCRC_STS SD Card Data CRC Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCSSN								DCSSP							
Type	RU								RU							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
15:8	**DCSSN**	DAT_CRCSTS_NEG	**MSDC DDR mode negative edge Read DATA CRC status** This register reflects the CRC status of data line[7:0] in DDR mode. The positive edge CRC status is shown in DAT_CRC_STS[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error
7:0	**DCSSP**	DAT_CRCSTS_POS	**MSDC read DATA CRC status** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error

11230070 EMMC_CFG0 EMMC Configuration Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**BTSUP**	**BTWDLY**											**BTACHKDIS**	**BTMOD**	**BTSTOP**	**BTSTART**
Type	RW	RW											RW	RW	WO	WO
Reset	0	0	0	0									0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**BTSUP**	BOOT_SUPPORT	**eMMC boot up support** The register bit indicates that boot mode is supported or not 1'b0: Not Support 1'b1: Support
14:12	**BTWDLY**	BOOT_WAIT_DELAY	**eMMC wait delay time** The register bit set the delay time to wait MMC device to exit boot up mode after boot stop bit is set. 3'b000: 0x1024 clock cycles 3'b001: 1x1024 clock cycles 3'b010: 2x1024 clock cycles 3'b111: 7x1024 clock cycles
3	**BTACHKDIS**	BOOT_ACK_CHK_DIS	**eMMC boot up mode ACK check Disable** 1'b0: Do ACK pattern check 1'b1: Bypass ACK pattern check
2	**BTMOD**	BOOT_MODE	**eMMC boot up mode** There are two kinds of boot up mode supported by eMMC 4.4. Reset CMD mode is option for eMMC 4.3. 1'b0: Pull low CMD mode 1'b1: Reset CMD mode
1	**BTSTOP**	BOOT_STOP	**eMMC boot up mode stop** The register bit is indicated that boot stop signal, read always return 0.
0	**BTSTART**	BOOT_START	**eMMC boot up start signal trigger** The register bit is boot up start signal trigger. read always return 0.

11230074 EMMC_CFG1 EMMC Configuration Register 1 00200003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BTATOC												BTDTOC[19:16]			
Type	RW												RW			
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BTDTOC[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:20	**BTATOC**	BOOT_ACK_TOC	**eMMC ack pattern time out counter** in unit of 2^16 serial clock. SW could not set it to 12'hFFF.
19:0	**BTDTOC**	BOOT_DAT_TOC	**eMMC read boot data time out counter** in unit of 2^16 serial clock. SW could not set it to 20'hFFFFF.

11230078 EMMC_STS EMMC Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										BTDRCV	BTARCV	BTSTS	BTATO	BTDTO	BTAERR	BTDERR
Type										RU	W1C	RU	W1C	W1C	W1C	W1C
Reset										0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
6	**BTDRCV**	BOOT_DAT_RECV	**eMMC boot data is received** This register is for S/W to check 1st 4-byte boot data is received or not. For other data after 1st 4B data, this bit should not be referenced. S/W should check RXFIFOCNT instead. 1'b0: There's no data in RXFIFO 1'b1: 1st 4B data is in RXFIFO
5	**BTARCV**	BOOT_ACK_RECV	**eMMC ack is received** Also need to check BOOT_ACK_ERR to determine pass or fail 1'b0: No ACK pattern is received 1'b1: ACK pattern has been received
4	**BTSTS**	BOOT_UP_STATE	**eMMC boot up mode status** The register bit indicates if MMC device operating in boot up mode state. 1'b0: Not in Boot up state 1'b1: Boot up state is on-going
3	**BTATO**	BOOT_ACK_TO	**eMMC ack timeout** The register bit indicates the controller detect a time out condition while waiting for an ack pattern on DAT0. 1'b0: No ACK pattern timeout error 1'b1: ACK pattern timeout error
2	**BTDTO**	BOOT_DAT_TO	**eMMC data timeout** The register bit indicates the controller detect a time out condition while waiting for boot data. 1'b0: No Data timeout error 1'b1: Data timeout error
1	**BTAERR**	BOOT_ACK_ERR	**eMMC ack error** The register bit indicates the status of ack pattern checking result.

Bit(s)	Mnemonic	Name	Description
			The bit is setting to 1 when ack pattern error. 1'b0: No ACK pattern check error 1'b1: ACK pattern check error
0	**BTDERR**	BOOT_CRC_ERR	**eMMC CRC error** The register bit indicates the CRC status of boot data. The bit is setting to 1 when data CRC error. 1'b0: No Data CRC error 1'b1: Data CRC error

1123007C EMMC_IOCON EMMC IO Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BTRST
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**BTRST**	BOOT_RST	**eMMC device boot up mode reset** The register bit is to trigger HW reset to set eMMC device entering into pre-idle state. 1'b0: de-assert RST_n to eMMC card 1'b1: Assert RST_n to eMMC card

11230080 SD_ACMD_RESP SD ACMD Response Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMDRESP[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMDRESP[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMDRESP**	AUTOCMD_RESP	**SD Auto command response register** This register stores the response[39:8] of ACMD12/ACMD23/ACMD19.

11230084 SD_ACMD19_TRG SD ACMD19 Target Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																

Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ACMDFTSEL			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**ACMDFTSEL**	FINE_TUNE_SEL	**SD Auto command 19 test target selection** After auto-command 19 is triggered, MSDC will only change the phase of the selected one and keep the value of other tuning registers. S/W can get the result from AUTOCMD_STS. There are total 32 phases for each delay line. 4'd0: Select PAD_CLK_TXDLY[5:0] as the target to run auto-command19. 4'd1: Select PAD_CMD_RXDLY[5:0] as the target to run auto-command19. 4'd2: Select PAD_DAT_RD_RXDLY[5:0] as the target to run auto-command19. 4'd3: Select PAD_DAT_WR_RXDLY[5:0] as the target to run auto-command19. 4'd4: Select DAT0_RD_DLY[5:0] as the target to run auto-command19. 4'd5: Select DAT1_RD_DLY[5:0] as the target to run auto-command19. 4'd6: Select DAT2_RD_DLY[5:0] as the target to run auto-command19. 4'd7: Select DAT3_RD_DLY[5:0] as the target to run auto-command19. 4'd8: Select DAT4_RD_DLY[5:0] as the target to run auto-command19. 4'd9: Select DAT5_RD_DLY[5:0] as the target to run auto-command19. 4'd10: Select DAT6_RD_DLY[5:0] as the target to run auto-command19. 4'd11: Select DAT7_RD_DLY[5:0] as the target to run auto-command19. 4'd12: Select CMD_RESP_RXDLY[5:0] as the target to run auto-command19. Others: Reserved

11230088 SD_ACMD19_STS SD ACMD19 Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD19STS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMD19STS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMD19STS**	AUTOCMD19_STS	**SD Auto command 19 test result register** When auto-command 19 is enabled, H/W will automatically try 32 times of command-19 and store the result into this register. This register contains 1st to 32th results in bit[0:31]

1123008C DMA_SA_HIGH4BIT DMA Current Address Resgiter of high 4bit 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													DMASAHIGH4BIT			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**DMASAHIGH4BIT**	DMA_SURR_ADDR_HIGH4BIT	**it is usend to set high 4bit address of start address because 64G dram need 36bit address**

11230090 DMA_SA DMA Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMASA[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMASA[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMASA**	DMA_STR_ADDR	**The start address of the DMA address** This register is used to set the start address of the DMA. In DMA basic mode, this field indicates the source or destination address of the data transfer which depends on the command. In descriptor base DMA, this is the descriptor chain start address.

11230094 DMA_CA DMA Current Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMACA**	DMA_CURR_ADDR	**The current address of the DMA address** This register is used to read the current address of the DMA descriptor chain.

11230098 DMA_CTRL DMA Control Register 00006008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		**BSTSZ**			**SPLIT1K**	**LASTBF**	**DMAALIGN**	**DMAMOD**					**READYM**	**DMARSM**	**DMASTOP**	**DMASTART**
Type		RW			RW	RW	RW	RW					RO	WO	A0	WO
Reset		1	1	0	0	0	0	0					1	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**BSTSZ**	BURST_SIZE	**DMA burst size** This field is used to specify the maximum transfer bytes allowed at the device per DMA burst. This field can not be modified when the DMA status is 1. 3'd3: 8 Bytes 3'd4: 16 Bytes 3'd5: 32 Bytes 3'd6: 64 Bytes Other: Reserved
11	**SPLIT1K**	DMA_SPLIT_1K	**This field is used to specify whether split burst when corss 1K boundry address** 1'b0: 1K boundary not split 1'b1: 1K boundary split
10	**LASTBF**	LAST_BUF	**Last buffer of the basic DMA mode** This field indicates the last buffer in the basic DMA mode
9	**DMAALIGN**	DMA_ALIGN	**This field is used to specify whether address aligment burst size** 1'b0: do not DAM burst size alignment 1'b1: DAM burst size alignment
8	**DMAMOD**	DMA_MODE	**DMA operation mode** This field indicates operation mode of DMA 1'b0: Basic DMA mode 1'b1: Descriptor base DMA mode
3	**READYM**	AHB_READYM	**only for debug when dma hang,sw can check if ahb bus is ok when gdma is hang** 1: bus is normal 0: bus not normal
2	**DMARSM**	DMA_RESUME	**DMA resume control register** This bit is used to resume the DMA transaction. Read always return 0
1	**DMASTOP**	DMA_STOP	**DMA Stop control register** This bit is used to stop the DMA transaction. When SW issue STOP command, SW must wait this bit de-assert or DMA inactive to guarantee stop done.
0	**DMASTART**	DMA_START	**DMA start control register** This bit is used to start the DMA transaction. Read always return 0

1123009C DMA_CFG DMA Configuration Register 00000004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															**OUTBOUND**	**DMACHKSU**

															STOPDMA	M12B
Type															RW	RW
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MSDCACTIVEEN				AHBHPROT2EN							LOCKDISABLE	DSCPCSEN	DMASTS
Type			RW				RW							RW	RW	RU
Reset			0	0			0	0						1	0	0

Bit(s)	Mnemonic	Name	Description
17	**OUTBOUNDSTOPDMA**	OUT_BOUND_STOP_DMA	**This register will determine whether stop Enhance DMA if gear setting is out-of-boundary during send training data** 1'b0: Enhance DMA will continue even if gear setting is out-of-boundary during send training data 1'b1: Enhance DMA will stop if gear setting is out-of boundary during send training data
16	**DMACHKSUM12B**	DMA_CHK_SUM_12B	**This register indicates GPD/BD checksum cover 16byte or 12byte** 1'b0: GPD/BD checksum cover 16byte 1'b1: GPD/BD checksum only cover 12byte
13:12	**MSDCACTIVEEN**	MSDC_ACTIVE_EN	**This register will indicate how to control msdc_active** 2'b00: dynamic control msdc_active 2'b01: msdc_active = 0 2'b10: msdc_active = 1 2'b11: Reserved
9:8	**AHBHPROT2EN**	AHB_HPROT_2_EN	**This register will determine how to control hprot_2 pin of AHB bus** AHB_HPROT_2_EN = 2'b00, and Basic DMA Mode All the write transfers of a burst will access by bufferable mode except the last burst of DMA AHB_HPROT2_2_EN=2'b00, and Descriptor DMA Mode all the write transfers of a burst will access by bufferable mode except HW own update transfer 2'b00: dynamic control hprot_2 2'b01: hprot_2 = 0 2'b10: hprot_2 = 1
2	**LOCKDISABLE**	LOCK_DISABLE	**should disable lock in order to improve emi efficient** 1'b0: enable ahb lock 1'b1: disable ahb lock
1	**DSCPCSEN**	DMA_DSCP_CS_EN	**DMA descriptor checksum enable** This bit is used to enable or disable the descriptor checksum validation function for the descriptor. This field can not be modified when the DMA status is 1.
0	**DMASTS**	DMA_STATUS	**DMA status** This bit is used to indicate the status of the DMA. 1'b0: DMA engine is inactive 1'b1: DMA engine is active

112300A0 SW_DBG_SEL MSDC S/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGSEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**SWDBGSEL**	DBG_SEL	**MSDC debug selection** This contain is reserved!

112300A4 SW_DBG_OUT MSDC S/W Debug Output Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SWDBGO[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGO[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SWDBGO**	DBG_OUT	**MSDC debug output** 32 bit output selected by SW_DBG_SEL register

112300A8 DMA_LENGTH DMA Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	XFSZ[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	XFSZ[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**XFSZ**	XFER_SIZE	**DMA total transfer size** This field is used to specify the number of DMA transfer byte required for the movement of source data through DMA. This field is only valid in basic DMA mode.

112300B0 PATCH_BIT0 MSDC Patch Bit Register 0 403C0006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PTCH 31	PTCH 30	PTCH 29	PTCH 28	PTCH 27	PTCH 26	PTCH22				PTCH18				PTCH 17	PTCH 16
Type	RW	RW	RW	RW	RW	RW	RW				RW				RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PTCH 15	CKGDLYS					INTCKS			DESC UP	PTCH 5	PTCH 4	PTCH 3	PTCH 02	PTCH 01	
Type	RW	RW					RW			RW	RW	RW	RW	RW	RW	

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	

Bit(s)	Mnemonic	Name	Description
31	**PTCH31**	EN_MMC_DRV_RESP	**Enable MSDC always drives bus when output wakeup response (BREAK)** 1'b0: Disable 1'b1: Enable
30	**PTCH30**	DETECT_WR_CRC_TIMEOUT	**MSDC write data CRC phase timeout detection** 1'b0: Not detect CRC phase timeout 1'b1: detect CRC phase timeout
29	**PTCH29**	SPC_ALWAYS_PUSH	**SPC Buffer push mechanism** 1'b0: Push the buffer only when read transfer is on-going 1'b1: Always push the buffer
28	**PTCH28**	SDIO_INT_DLY_SEL	**SDIO interrupt latch time selection** 1'b0: Latch the data line value in internal SDIO interrupt period 1'b1: Latch the data line value in 1 clock delay of internal SDIO interrupt period
27	**PTCH27**	SDC_CMD_CMDFAIL_SEL	**SDIO interrupt period recovery selection** 1'b0: SDIO interrupt period will re-start after a CMD12 or CMD52 command is issued 1'b1: SDIO interrupt period whenever DAT line is not busy
26	**PTCH26**	SDC_CMD_IDRT_SEL	**SD identification response time selection** The register bit indicates if the command has a response with NID (that is, 5 serial clock cycles as defined in SD Memory Card Specification Part 1 Physical Layer Specification version 1.0) response time. The register bit is valid only when the command has a response token. Thus the register bit must be set to 1 for CMD2 (ALL_SEND_CID) and ACMD41 (SD_APP_OP_CMD). 1'b0: Otherwise. 1'b1: The command has a response with NID response time.
25:22	**PTCH22**	SDC_CFG_WDOD	**SD Write Data Output Delay** The period from finish of the response for the initial host write command or the last write data block in a multiple block write operation to the start bit of the next write data block requires at least two serial clock cycles. The register field is used to extend the period (Write Data Output Delay) in unit of one serial clock. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
21:18	**PTCH18**	SDC_CFG_BSYDLY	**SD R1B busy detection mode** The register field is only valid for the commands with R1b response. If the command has a response of R1b type, MS/SD controller must monitor the data line 0 for card busy status from the bit time that is two serial clock cycles after the command end bit to check if operations in SD/MMC Memory Card have finished. The register field is used to expand the time between the command end bit and end of detection period to detect card busy status. If time is up and there is no card busy status on data line 0, then the controller will abandon the detection. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
17	**PTCH17**	SDIO_CFG_INTC_SEL	**SDIO Interrupt model selection** 1'b0: Only when data line [1] = 0 and then trigger SDIO interurpt event

Bit(s)	Mnemonic	Name	Description
			1'b1: Only when data line [3:0] = 4'b1101 and then trigger SDIO interurpt event
16	**PTCH16**	MSDC_BLKNUM_SEL	**Configuration support ACMD23 reliable/force prog etc. feature** 1'b0: Support ACMD23 reliable/force prog etc. feature 1'b1: Don't support ACMD23 reliable/force prog etc. feature
15	**PTCH15**	MSDC_FIFO_RD_DIS	**MSDC RXFIFO Read Disable** 1'b0: Disable FIFO read permission to RXFIFO in PIO mode 1'b1: Enable FIFO read permission to RXFIFO in PIO mode
14:10	**CKGDLYS**	CKGEN_MSDC_DLY_SEL	**CKBUF in CKGEN Delay Selection** Total 32 stages
9:7	**INTCKS**	INT_DAT_LATCH_CK_SEL	**Internal MSDC clock phase selection** Total 8 stages, each stage can delay 1 clock period of msdc_src_ck
6	**DESCUP**	DESC_UP_SEL	**sd transfer done int should be issue when GPD have been update** 1'b1: enable new function for generate sd transfer done int 1'b0: use old function for generate sd transfer done int
5	**PTCH5**	ACMD53_FAIL_ONE_SHOT	**determine interrupt method of AUTOCMD53_FAIL** 1'b0: AUTOCMD53_FAIL interrupt will assert whenever CMD/DAT crc error occur 1'b1: AUTOCMD53_FAIL interrupt will assert only when AUTOCMD53_DONE assert if there is CMD/DAT crc error
4	**PTCH4**	MASK_ACMD53_CRC_ERR_INTR	**mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence** 1'b0: enable CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence 1'b1: mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence
3	**PTCH3**	RD_DAT_SEL	**This field is used to define whther used rising or falling buf data for SDR mode** 1'b0: Used rising buf data for SDR mode 1'b1: Used falling buf data for SDR mode
2	**PTCH02**	DIS_REFLECT_CMDWR_WHEN_BSY	**Enable SD command register write montior** 1'b0: Enable monitor function 1'b1: Disable monitor function
1	**PTCH01**	EN_SDC_ODD_8BIT_SUP	**Enable SD odd number support for 8-bit data bus** 1'b0: Disable 1'b1: Enable

112300B4 **PATCH_BIT1** **MSDC Patch Bit Register 1** **FFB00009**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSHBFCKEN	MRCTLCKEN	MWCTLCKEN	MSDCKEN	MACMDCKEN	MVOLDTCKEN	MPSCCKEN	MSPCCKEN	HGDMACKEN		DCMEN	DCMDIVSEL1				SINGLEBURST
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW				RW
Reset	1	1	1	1	1	1	1	1	1		1	1				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESETGDMA		BIAS28R0	BIAS28R1	BIAS28R2				GETCRCMARGIN	GETBUSYMARGIN	CMDTA			WRTA		
Type	RW		RW	RW	RW				RW	RW	RW			RW		
Reset	0		0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
31	**MSHBFCKEN**	MSDC_CK_SHBFF_CKEN	**msdc_src_ck clock enable bit for SHBFF** 1'b0: Disable 1'b1: Enable
30	**MRCTLCKEN**	MSDC_CK_RCTL_CKEN	**msdc_src_ck clock enable bit for RCTL** 1'b0: Disable 1'b1: Enable
29	**MWCTLCKEN**	MSDC_CK_WCTL_CKEN	**msdc_src_ck clock enable bit for WCTL** 1'b0: Disable 1'b1: Enable
28	**MSDCKEN**	MSDC_CK_SD_CKEN	**msdc_src_ck clock enable bit for SD** 1'b0: Disable 1'b1: Enable
27	**MACMDCKEN**	MSDC_CK_ACMD_CKEN	**msdc_src_ck clock enable bit for ACMD** 1'b0: Disable 1'b1: Enable
26	**MVOLDTCKEN**	MSDC_CK_VOLDET_CKEN	**msdc_src_ck clock enable bit for VOLDET** 1'b0: Disable 1'b1: Enable
25	**MPSCCKEN**	MSDC_CK_PSC_CKEN	**msdc_src_ck clock enable bit for PSC** 1'b0: Disable 1'b1: Enable
24	**MSPCCKEN**	MSDC_CK_SPC_CKEN	**msdc_src_ck clock enable bit forSPC** 1'b0: Disable 1'b1: Enable
23	**HGDMACKEN**	AHB_CK_GDMA_CKEN	**hclk_ck clock enable bit for GDMA** 1'b0: Disable 1'b1: Enable
21	**DCMEN**	DCM_EN	**host enable dcm for low power,when there is no transfer the ahb_ck and msdc_src_ck will divder from spurce clk** 1'b1: disable 1'b0: enable
20	**DCMDIVSEL1**	DCM_DIV_SEL1	**for spec4.5 divider msdc_src_ck from source clk,for spec50 divider ahb_ck from source clk** 1'b1: 32 divder from source clk 1'b0: 16 divder from source clk
16	**SINGLEBURST**	ENABLE_SINGLE_BURST	**the ahb bus will not support incr1 burst type in future.And it will only affect AHB bus msdc design,not affect AXI bus design** 1'b0: hw will send incr1 burst type 1'b1: hw will send single burst typr instead of incr1 type
15	**RESETGDMA**	RESET_GDMA	**sw can sw reset gdma when design hang** 1 'b1: reset gdma 1'b0: not reset gdma
13	**BIAS28R0**	BIAS_EXTBIAS_28NM	**28NM BIAS Controler register 0**
12	**BIAS28R1**	BIAS_EN18IO_28NM	**28NM BIAS Controler register 1**
11:8	**BIAS28R2**	BIAS_TUNE_28NM	**28NM BIAS Controler register 2**
7	**GETCRCMARGIN**	GET_CRC_MARGIN	**it will add margin for get crc status when card resp crc not match spec 2cycle from endbit** 1'b0: 8 cycle reserved for get crc status from write data crc endbit 1'b1: 16 cycle reserved for get crc status from write data crc endbit
6	**GETBUSYM**	GET_BUSY_MARG	**it will add margin for get busy state of data0**

Bit(s)	Mnemonic	Name	Description
	ARGIN	IN	1'b0: 1 cycle reserved for get busy state from src status endbit 1'b1: 3cycle reserved for get busy state from src status endbit
5:3	**CMDTA**	CMD_RSP_TA_CNTR	**CMD response turn around period** The turn around cycle = CMD_RSP_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode, this register should be set to 0
2:0	**WRTA**	WRDAT_CRCS_TA_CNTR	**Write data and CRC status turn around period** The turn around cycle = WRDAT_CRCS_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode,this register should be set to 0

112300B8 PATCH_BIT2 MSDC Patch Bit Register 2 14801803

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCSTSENSEL			CFGCRCSTS	CFGCRCSTSCNT		CFGCRCSTSEDGE	CFG_CRCSTS_SEL	POPENCNT					RESPSTSENSEL		
Type	RW			RW	RW		RW	RW	RW					RW		
Reset	0	0	0	1	0	1	0	0	1	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFGRESP	CFGRESPCNT			INTCRESPSEL		CFGRDAT	CFGRDATCNT					RESPWAITCNT		SUPPORT64G	ENHANCEGPD
Type	RW	RW			RW		RW	RW					RW		RW	RW
Reset	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:29	**CRCSTSENSEL**	CRCSTS_LATCH_EN_SEL	**This register used configuration latch CRC Status enable signal for async fifo in emmc45** 3'b000: latch CRC Status enable signal not delay 3'b001: latch CRC Status enable signal delay 1T msdc_ck 3'b010: latch CRC Status enable signal delay 2T msdc_ck 3'b011: latch CRC Status enable signal delay 3T msdc_ck 3'b111: latch CRC Status enable signal delay 7T msdc_ck
28	**CFGCRCSTS**	CFG_CRCSTS	**This register used configuration CRC Status path selection,this setting only used emmc4.5 feature** 1'b0: Latch CRC Status select delay-line path 1'b1: Latch CRC Status select async fifo path
27:26	**CFGCRCSTSCNT**	CFG_CRCSTS_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 2'b00: push 0 data in async fifo when start pop out data from async fifo 2'b01: push 1 data in async fifo when start pop out data from async fifo 2'b10: push 2 data in async fifo when start pop out data from async fifo 2'b11: push 3 data in async fifo when start pop out data from async fifo
25	**CFGCRCSTSEDGE**	CFG_CRCSTS_EDGE	**This register configuration used rising async fifo or falling async fifo** 1'b0: async fifo latch CRC Status used rising async fifo 1'b1: async fifo latch CRC Status used falling async fifo

Bit(s)	Mnemonic	Name	Description
24		CFG_CRCSTS_SEL	**This register configuration async fifo path selection** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
23:20	**POPENCNT**	POP_EN_CNT	**pop enable counter** This field is used to define how many write pointer and the read pointer margin began to pop data transfer
18:16	**RESPSTSENSEL**	RESP_LATCH_EN_SEL	**This register used configuration latch CMD Response enable signal for async fifo in emmc45** 3'b000: latch CMD Response enable signal not delay 3'b001: latch CMD Response enable signal delay 1T msdc_ck 3'b010: latch CMD Response enable signal delay 2T msdc_ck 3'b011: latch CMD Response enable signal delay 3T msdc_ck 3'b111: latch CMD Response enable signal delay 7T msdc_ck
15	**CFGRESP**	CFG_RESP	**This register used configuration CMD Response path selection,this setting only used emmc4.5 feature** 1'b0: Latch CMD Response select async fifo path 1'b1: Latch CMD Response select delay-line path
14:12	**CFGRESPCNT**	CFG_RESP_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 3'b000: push 0 data in async fifo when start pop out data from async fifo 3'b001: push 1 data in async fifo when start pop out data from async fifo 3'b111: push 7 data in async fifo when start pop out data from async fifo
11	**INTCRESPSEL**	INTC_RESP_SEL	**This register configuration BREAK command async fifo path** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
9	**CFGRDAT**	CFG_RDAT	**This register used configuration read data path** 1'b0: read data path by pass delay line 1'b1: read data path through delay line
8:4	**CFGRDATCNT**	CFG_RDAT_CNT	**This register used configuration read data path delay line**
3:2	**RESPWAITCNT**	RESP_WAIT_CNT	**This register used configuration cmd response timeout,timeout cycle is 65T+16*RESP_WAIT_CNT** 2'b00: cmd response timeout is 65T 2'b01: cmd response timeout is 65T+ 16*1T 2'b01: cmd response timeout is 65T+ 16*2T 2'b01: cmd response timeout is 65T+ 16*3T
1	**SUPPORT64G**	SUPPORT_64G	**This register used which proj support high 64G dram space access** 1'b1: support 64G dram access 1'b0: not support 64G dram access
0	**ENHANCEGPD**	ENHANCE_WAIT_GPD	**if sw clear int when gpd not update finish,design will hang.so you can set this bt to 1 to avoid this iuuse in enhance write mode** 1'b1: use new HW code for update gpd in enhance mode 1'b0: use old HW code

112300C0 DAT0_TUNE_C DAT0 Tune Result Register 00000000

RC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT0CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT0CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT0CRCSTS**	DAT0_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT0 into this register. This register contains 1st to 32th results in bit[0:31]

112300C4 DAT1_TUNE_CRC DAT1 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT1CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT1CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT1CRCSTS**	DAT1_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT1 into this register. This register contains 1st to 32th results in bit[0:31]

112300C8 DAT2_TUNE_CRC DAT2 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT2CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT2CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT2CRCSTS**	DAT2_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT2 into this register.

Bit(s)	Mnemonic	Name	Description
			This register contains 1st to 32th results in bit[0:31]

112300CC **DAT3_TUNE_CRC** **DAT3 Tune Result Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT3CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT3CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT3CRCSTS**	DAT3_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT3 into this register. This register contains 1st to 32th results in bit[0:31]

112300D0 **CMD_TUNE_CRC** **CMD Tune Result Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CMDCRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMDCRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CMDCRCSTS**	CMD_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of CMD into this register. This register contains 1st to 32th results in bit[0:31]

112300D4 **SDIO_TUNE_WIND** **SDIO Tune Window Register 0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												TUNEWINDOW				
Type												RW				
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:0	**TUNEWINDOW**	TUNE_WINDOW	**SDIO Auto command 53 tuning window setting. the window range will be** 1. for CMD PAD low bound: max(0, PAD_CMD_RXDLY-TUNE_WINDOW) high bound: min(31, PAD_CMD_RXDLY+TUNE_WINDOW) 2. for DAT PAD low bound: max(0, DAT(0:3)_RD_DLY-TUNE_WINDOW) high bound:min(31: DAT(0:3)_RD_DLY+TUNE_WINDOW)

112300F0 PAD_TUNE0 MSDC Pad Tuning Register0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKTDLY					CMDRRDLY					CMDRRDLYSEL	CMDRDLY				
Type	RW					RW					RW	RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDLYSEL		DATRRDLYSEL	DATRRDLY					DELAYEN			DATWRDLY				
Type	RW		RW	RW					RW			RW				
Reset	0		0	0	0	0	0	0	0			0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:27	**CLKTDLY**	PAD_CLK_TXDLY	**CLK Pad TX Delay Control** This register is used to add delay to CLK phase. Total 32 stages
26:22	**CMDRRDLY**	PAD_CMD_RESP_RXDLY	**CMD Response Internal Delay Line Control** This register is used to fine-tune response phase latched by MSDC internal clock Total 32 stages
21	**CMDRRDLYSEL**	PAD_CMD_RD_RXDLY_SEL	**Decide CMD Response pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY**	PAD_CMD_RXDLY	**CMD Pad RX Delay Line1 Control** This register is used to fine-tune CMD pad macro respose latch timing Total 32 stages
15	**RXDLYSEL**	PAD_RXDLY_SEL	**Decide rx delay line tune data path or clock path** 1'b0: rx delay line tune data path 1'b1: rx delay line tune clock path
13	**DATRRDLYSEL**	PAD_DAT_RD_RXDLY_SEL	**Decide rx data pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY**	PAD_DAT_RD_RXDLY	**DAT Pad RX Delay Line1 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages
7	**DELAYEN**	DELAY_EN	**enable all delay cell toogle when power on** 1'b0: disable delay cell toogle default 1'b1: enable delay cell toogle default

Bit(s)	Mnemonic	Name	Description
4:0	**DATWRDLY**	PAD_DAT_WR_RXDLY	**Write Data Status Internal Delay Line Control** This register is used to fine-tune write status phase latched by MSDC internal clock Total 32 stages

112300F4 PAD_TUNE1 MSDC Pad Tuning Register1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											CMDRRDLY2SEL	CMDRDLY2				
Type											RW	RW				
Reset											0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DATRRDLY2SEL	DATRRDLY2												
Type			RW	RW												
Reset			0	0	0	0	0	0								

Bit(s)	Mnemonic	Name	Description
21	**CMDRRDLY2SEL**	PAD_CMD_RD_RXDLY2_SEL	**Decide CMD Response pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY2**	PAD_CMD_RXDLY2	**CMD Pad RX Delay Line2 Control** This register is used to fine-tune CMD pad macro respose latch timing in data path Total 32 stages
13	**DATRRDLY2SEL**	PAD_DAT_RD_RXDLY2_SEL	**Decide rx data pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY2**	PAD_DAT_RD_RXDLY2	**DAT Pad RX Delay Line2 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages

112300F8 DAT_RD_DLY0 MSDC Data Delay Line Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY								DAT1RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY								DAT3RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY**	DAT0_RD_DLY	**DAT0 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY**	DAT1_RD_DLY	**DAT1 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

Bit(s)	Mnemonic	Name	Description
12:8	**DAT2RDDLY**	DAT2_RD_DLY	**DAT2 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY**	DAT3_RD_DLY	**DAT3 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

112300FC DAT_RD_DLY1 MSDC Data Delay Line Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY								DAT5RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY								DAT7RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY**	DAT4_RD_DLY	**DAT4 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY**	DAT5_RD_DLY	**DAT5 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY**	DAT6_RD_DLY	**DAT6 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY**	DAT7_RD_DLY	**DAT7 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

11230100 DAT_RD_DLY2 MSDC Data Delay Line Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY2								DAT1RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY2								DAT3RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY2**	DAT0_RD_DLY2	**DAT0 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY2**	DAT1_RD_DLY2	**DAT1 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT2RDDLY2**	DAT2_RD_DLY2	**DAT2 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY2**	DAT3_RD_DLY2	**DAT3 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11230104 DAT_RD_DLY3 MSDC Data Delay Line Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY2								DAT5RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY2								DAT7RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY2**	DAT4_RD_DLY2	**DAT4 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY2**	DAT5_RD_DLY2	**DAT5 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY2**	DAT6_RD_DLY2	**DAT6 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY2**	DAT7_RD_DLY2	**DAT7 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11230110 HW_DBG_SEL MSDC H/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBGWSEL	DBG3SEL						DBG2SEL							
Type		RW	RW						RW							
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DBG1SEL						DBG0SEL							
Type			RW						RW							
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30	**DBGWSEL**	HW_DBG_WRAP_SEL	**H/W debug output selection for wrapper** 0: Select original debug pins 1: Select wrapper debug pins
29:24	**DBG3SEL**	HW_DBG3_SEL	**H/W debug output selection**
23:16	**DBG2SEL**	HW_DBG2_SEL	**H/W debug output selection**
13:8	**DBG1SEL**	HW_DBG1_SEL	**H/W debug output selection**
7:0	**DBG0SEL**	HW_DBG0_SEL	**H/W debug output selection**

11230114 MAIN_VER MSDC Main Version Register 20140512

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINVER[31:16]															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAINVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:0	**MAINVER**	MAIN_VER	**Main Version**

11230118 ECO_VER MSDC ECO Version Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECOVER[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECOVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ECOVER**	ECO_VER	**ECO Version**

Module name: MSDC Base address: (+0h)

Address	Name	Width	Register Function
11240000	**MSDC_CFG**	32	**MSDC Configuration Register** The register is for general configuration of the MS/SD controller.
11240004	**MSDC_IOCON**	32	**MSDC IO Configuration Register** The register contains the receiver path data latch timing control and interface control bits.
11240008	**MSDC_PS**	32	**MSDC Pin Status Register** The register is used to storing card detection and write protection pin status. Card detection status can be disabled.
1124000C	**MSDC_INT**	32	**MSDC Interrupt Register** The register contains the status of interrupts. Note that the register still shows the status of interrupt even though the interrupt is disabled.
11240010	**MSDC_INTEN**	32	**MSDC Interrupt Enable Register** The register contains the related enable bit of interrupts.
11240014	**MSDC_FIFOCS**	32	**MSDC FIFO Control and Status Register** The register contains the control and status of embedded 128B FIFO.
11240018	**MSDC_TXDATA**	32	**MSDC TX Data Port Register** The register is for PIO mode only. Used to input MSDC write data to card. The access can be AHB 1B/2B/4B
1124001C	**MSDC_RXDATA**	32	**MSDC RX Data Port Register** The register is for PIO mode only. Used to read back MSDC read data from card. The access can be AHB 1B/2B/4B.
11240030	**SDC_CFG**	32	**SD Configuration Register** The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
11240034	**SDC_CMD**	32	**SD Command Register** The register defines a SD Memory Card command and its attributes. Before MS/SD controller issues a transaction onto SD

			bus, application shall specify other relative settings such as argument for command. After writing the register by the application, MS/SD controller will issue the corresponding transaction onto SD serial bus. If the command is GO_IDLE_STATE, the controller will have serial clock on SD/MMC bus run 128 cycles before issuing the command.
11240038	**SDC_ARG**	32	**SD Argument Register** The register contains the argument of the SD/MMC Memory Card command.
1124003C	**SDC_STS**	32	**SD Status Register** The register reflects SD bus status and contains MMC stream write status.
11240040	**SDC_RESP0**	32	**SD Response Register 0** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11240044	**SDC_RESP1**	32	**SD Response Register 1** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11240048	**SDC_RESP2**	32	**SD Response Register 2** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
1124004C	**SDC_RESP3**	32	**SD Response Register 3** The register contains parts of the last SD/MMC Memory Card bus response. The register fields SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3 are composed of the last SD/MMC Memory card bus response. For response of type R2, that is, response of the command ALL_SEND_CID, SEND_CSD and SEND_CID, only bit 127 to 0 of response token is stored in the register field SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3. SDC_RESP0 = bit 31~0 SDC_RESP1 = bit 63~32 SDC_RESP2 = bit 95~64 SDC_RESP3 = bit 127~96 For response of type R1b in auto CMD12 or R1 in auto CMD23, bit 39 to 8 of response token is stored in the register field of SDC_RESP3. For the responses of other types, only bit 39 to 8 of response token is stored in the register field SDC_RESP0.
11240050	**SDC_BLK_NUM**	32	**SD Block Number Register** This register defines the block number for the block transaction. For single read/write, this register should be set to 1. For multiple read/write, this register should be set to larger than 1. Set to 0 will cause unexpected result.
11240054	**SDC_VOL_CHG**	32	**SD Voltage Change Wait Time Register** This register define SD voltage change check wait time
11240058	**SDC_CSTS**	32	**SD Card Status Register** After commands with R1 and R1b response, this register will contain the status of the SD/MMC card
1124005C	**SDC_CSTS_EN**	32	**SD Card Status Enable Register** This register is used to control which bit of the SDC_CSTS will generate the MSDC_INT.SD_CSTA interrupt.
11240060	**SDC_DATCRC_STS**	32	**SD Card Data CRC Status Register** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read
11240080	**SD_ACMD_RESP**	32	**SD ACMD Response Register** This register stores the response of auto command from SD card
11240084	**SD_ACMD19_TRG**	32	**SD ACMD19 Target Register** This register is used to select target delay line to run ACMD19 sequence.

11240088	SD_ACMD19_STS	32	**SD ACMD19 Status Register** This register stores the result of auto command 19 from SD card
1124008C	DMA_SA_HIGH4BIT	32	**DMA Current Address Resgiter of high 4bit** This register contain the start address high 4bit of 36bit address for 64G dram access
11240090	DMA_SA	32	**DMA Start Address Register** This register contains the start address of the DMA descriptor
11240094	DMA_CA	32	**DMA Current Address Register** This register contains the current DMA address
11240098	DMA_CTRL	32	**DMA Control Register** This register is used to control the DMA operation.
1124009C	DMA_CFG	32	**DMA Configuration Register** This register is used to configure the DMA operation.
112400A0	SW_DBG_SEL	32	**MSDC S/W Debug Selection Register** This register is used to select S/W debug output
112400A4	SW_DBG_OUT	32	**MSDC S/W Debug Output Register** This register shows the selected debug output
112400A8	DMA_LENGTH	32	**DMA Length Register** This register is used to set Basic DMA operation length
112400B0	PATCH_BIT0	32	**MSDC Patch Bit Register 0** This register can configure the patch function. For normal function, these bit should keep in default value
112400B4	PATCH_BIT1	32	**MSDC Patch Bit Register 1** This register can configure the patch function. For normal function, these bit should keep in default value
112400B8	PATCH_BIT2	32	**MSDC Patch Bit Register 2** This register can configure the patch function. For normal function, these bit should keep in default value
112400C0	DAT0_TUNE_CRC	32	**DAT0 Tune Result Register** This register record on-line tuning result for DAT0 line
112400C4	DAT1_TUNE_CRC	32	**DAT1 Tune Result Register** This register record on-line tuning result for DAT1 line
112400C8	DAT2_TUNE_CRC	32	**DAT2 Tune Result Register** This register record on-line tuning result for DAT2 line
112400CC	DAT3_TUNE_CRC	32	**DAT3 Tune Result Register** This register record on-line tuning result for DAT3 line
112400D0	CMD_TUNE_CRC	32	**CMD Tune Result Register** This register record on-line tuning result for CMD line
112400D4	SDIO_TUNE_WIND	32	**SDIO Tune Window Register 0** This register define tuning window size for SDIO on-line CRC tuning feature
112400F0	PAD_TUNE0	32	**MSDC Pad Tuning Register0** This register can configure the delay line embedded in Pad Macro
112400F4	PAD_TUNE1	32	**MSDC Pad Tuning Register1** This register can configure the delay line embedded in Pad Macro
112400F8	DAT_RD_DLY0	32	**MSDC Data Delay Line Register 0** This register can configure the delay line embedded in Pad Macro
112400FC	DAT_RD_DLY1	32	**MSDC Data Delay Line Register 1** This register can configure the delay line embedded in Pad Macro
11240100	DAT_RD_DLY2	32	**MSDC Data Delay Line Register 2** This register can configure the delay line embedded in Pad Macro
11240104	DAT_RD_DLY3	32	**MSDC Data Delay Line Register 3** This register can configure the delay line embedded in Pad Macro

11240110	**HW_DBG_SEL**	32	**MSDC H/W Debug Selection Register** This register can select the H/W debug output
11240114	**MAIN_VER**	32	**MSDC Main Version Register** This register shows the version code of MSDC IP
11240118	**ECO_VER**	32	**MSDC ECO Version Register** This register shows the ECO version code of MSDC IP

11240000 MSDC_CFG MSDC Configuration Register 02000099

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							SCLKSTOPSEL				CCKMD		CCKDIV[11:8]			
Type							RW				W1C		RW			
Reset							1				0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCKDIV[7:0]								CCKSB	BV18PSS	BV18SDT	CCKDRVE	PIO	RST	CCKPD	MSDC
Type	RW								RU	RU	RW	RW	RW	A0	RW	RW
Reset	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1

Bit(s)	Mnemonic	Name	Description
25	**SCLKSTOPSEL**	SCLK_STOP_SEL	**In DDR mode,stop SCLK when device is idle whether check SCLK phase** 1'b0: stop sclk no check SCLK phase 1'b1: stop sclk check SCLK phase,fix 1/4T sclk glitch in DDR mode
21:20	**CCKMD**	CARD_CK_MODE	**MS/SD Card clock mode** 2'b00: Use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b01: Use msdc_src_ck as msdc_ck, bit[15]~bit[8] is ignored. 2'b10: DDR mode, also use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b11: HS400 mode,also use clock divider output and use msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed.
19:8	**CCKDIV**	CARD_CK_DIV	**MS/SD Card clock divider** The register field controls clock frequency of serial clock on MS/SD bus. Please refer to Data Line Latching Timing Diagram and Response Latching Timing Diagram. For non-DDR mode, msdc_ck equals SD bus clock. (Ex: For SDR25 or HS, msdc_ck and SD bus clock will be 50MHz) For DDR mode, msdc_ck denotes the MSDC internal clock which will be dobule to SD bus clock. (Ex: For DDR50, msdc_ck should be set to 100MHz and bus clock will be 50MHz) 8'b00000000: msdc_ck =(1/2) * msdc_src_ck 8'b00000001: msdc_ck = (1/(4*1)) * msdc_src_ck 8'b00000010: msdc_ck = (1/(4*2)) * msdc_src_ck 8'b00000011: msdc_ck = (1/(4*3))* msdc_src_ck 8'b00010000: msdc_ck = (1/(4*16))* msdc_src_ck 8'b11111111: msdc_ck = (1/(4*255)) * msdc_src_ck
7	**CCKSB**	CARD_CK_STABLE	**MS/SD Card clock stable or not** After programming the CARD_CK_MODE or CARD_CK_DIV, this bit will immediately go to "0" and return to "1" if stable. User should poll this register to make sure the safety control of MSDC. 1'b0: Clock output is not stable 1'b1: Clock output is stable
6	**BV18PSS**	BV_1P8_PASS	**MSDC Bus voltage 1.8V detection status** S/W should check this bit after BUS_VOL_18V_START_DET turns

Bit(s)	Mnemonic	Name	Description
			to 0 from 1. 1'b0: The voltage detection has error. 1'b1: The voltage detection has no error.
5	**BV18SDT**	BV_1P8_START_DET	**MSDC Bus voltage 1.8V detection sequence start event** S/W writes this bit to 1 to trigger H/W outputs 1.8V clock for 1 ms and automatically detect CMD/DAT line sequence for voltage change is passed or not. H/Q will clear this bit to 0 after the detection has finished. The pass or fail status is stored in bit[6] BUS_VOL_18V_PASS.
4	**CCKDRVE**	CARD_CK_DRV_EN	**SD/MS Card Bus Clock drive enable bit** Set this bit to 1 to enable MSDC bus clock driver. The default bus state depends on MSDC_CFG][1] CARD_CK_PWDN bit. If MSDC_CFG[1] CARD_CK_PWDN= 1, the default clock state is free running. If MSDC_CFG[1] CARD_CK_PWDN = 0, the default clock state is gated to 0. Set this bit to 0 will put the bus state into "tri-state". Default is 1. 1'b0: Put the clock pad into tri-state 1'b1: Enable MSDC to drive clock pad, the state of CLK depends on MSDC_CFG[1] CARD_CK_PWDN
3	**PIO**	PIO_MODE	**MS/SD PIO mode** PIO mode selection. Default is in PIO mode. 1'b0: DMA mode 1'b1: PIO mode
2	**RST**	RST	**Software reset** Writing 1 to this register will cause internal synchronous reset of MS/SD controller, and it will not reset register settings and DMA controller. The reset sequence is done when this bit goes to 0. S/W should wait this bit back to 0 after writing 1. 1'b0: MS/SD controller is not in reset state 1'b1: MS/SD controller is in reset state
1	**CCKPD**	CARD_CK_PWDN	**MSDC bus clock power down mode** This bit controls the card clock power down mode. 1'b0: Clock is gated to 0 if no command or data is transmitted. 1'b1: Clock is free running even if no command or data is transmitted. (The clock may still be stopped when MSDC write data is not enough or no space for next read data)
0	**MSDC**	MSDC	**MS/SD mode selection** The register bit is used to configure the controller as the host of Memory Stick or as the host of SD/MMC Memory card. The default value is to configure the controller as the host of Memory Stick. 1'b0: Configure the controller as the host of Memory Stick 1'b1: Configure the controller as the host of SD/MMC Memory card

11240004 MSDC_IOCON MSDC IO Configuration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									RD7SPL	RD6SPL	RD5SPL	RD4SPL	RD3SPL	RD2SPL	RD1SPL	RD0SPL
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			WD3S	WD2S	WD1S	WD0S	WDSP	WDSP			RDSP	DDR5	DDLS	RDSP	RSPL	SDR1

			PL	PL	PL	PL	LSEL	L			LSEL	0CKD	EL	L		04CKS
Type			RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23	**RD7SPL**	R_D7_SMPL	**Read data 7 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
22	**RD6SPL**	R_D6_SMPL	**Read data 6 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
21	**RD5SPL**	R_D5_SMPL	**Read data 5 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
20	**RD4SPL**	R_D4_SMPL	**Read data 4 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
19	**RD3SPL**	R_D3_SMPL	**Read data 3 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
18	**RD2SPL**	R_D2_SMPL	**Read data 2 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
17	**RD1SPL**	R_D1_SMPL	**Read data 1 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
16	**RD0SPL**	R_D0_SMPL	**Read data 0 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
13	**WD3SPL**	W_D3_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
12	**WD2SPL**	W_D2_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
11	**WD1SPL**	W_D1_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
10	**WD0SPL**	W_D0_SMPL	**CRC Status and SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock

Bit(s)	Mnemonic	Name	Description
			falling edge
9	**WDSPLSEL**	W_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in write transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.W_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.W_D0_SMPL Data line 1: MSDC_IOCON.W_D1_SMPL Data line 2: MSDC_IOCON.W_D2_SMPL Data line 3: MSDC_IOCON.W_D3_SMPL
8	**WDSPL**	W_D_SMPL	**CRC Status and SDIO interrupt sample selection** 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
5	**RDSPLSEL**	R_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in read transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.R_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.R_D0_SMPL Data line 1: MSDC_IOCON.R_D1_SMPL Data line 2: MSDC_IOCON.R_D2_SMPL Data line 3: MSDC_IOCON.R_D3_SMPL Data line 4: MSDC_IOCON.R_D4_SMPL Data line 5: MSDC_IOCON.R_D5_SMPL Data line 6: MSDC_IOCON.R_D6_SMPL Data line 7: MSDC_IOCON.R_D7_SMPL
4	**DDR50CKD**	DDR50_DLY_SEL	**DDR50 output clock delay selection** 1'b0: Use default clock output 1'b1: Delay 1T msdc_src_ck for clock output
3	**DDLSEL**	D_DLYLINE_SEL	**Data line delay line fine tune selection** 1'b0: All data line share one delay selection value indicated by PAD_TUNE.PAD_DAT_RD_RXDLY 1'b1: Each data line has its own delay selection value indicated by Data line 0: DAT_RD_DLY0.DAT0_RD_DLY Data line 1: DAT_RD_DLY0.DAT1_RD_DLY Data line 2: DAT_RD_DLY0.DAT2_RD_DLY Data line 3: DAT_RD_DLY0.DAT3_RD_DLY Data line 4: DAT_RD_DLY1.DAT4_RD_DLY Data line 5: DAT_RD_DLY1.DAT5_RD_DLY Data line 6: DAT_RD_DLY1.DAT6_RD_DLY Data line 7: DAT_RD_DLY1.DAT7_RD_DLY
2	**RDSPL**	R_D_SMPL	**Read data sample selection** 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
1	**RSPL**	R_SMPL	**Command response sample selection** 1'b0: Sample response by external bus clock rising edge 1'b1: Sample response by external bus clock falling edge
0	**SDR104CKS**	SDR104_CLK_SEL	**SDR104 SCLK output clock control** This bit is only used when MSDC_CFG[17:16] CARD_CK_MODE is 2'b01. 1'b0: Bus clock output equals inverted msdc_src_ck 1'b1: Bus clock output equals msdc_src_ck

11240008 MSDC_PS MSDC Pin Status Register 01FF0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SDWP							CMD	DAT							
Type	RU							RU	RU							
Reset	0							1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CDDBCE														CDSTS	CDEN
Type	RW														RU	RW
Reset	0	0	0	0											1	0

Bit(s)	Mnemonic	Name	Description
31	**SDWP**	SD_WP	**Write Protection Switch status on SD Memory Card** The register bit shows the status of Write Protection Switch on SD Memory Card. There is no default reset value. The pin WP (Write Protection) is only useful while the controller is configured for SD Memory Card 1'b0: Write Protection Switch ON. It means that memory card is desired to be write-protected 1'b1: Write Protection Switch OFF. It means that memory card is writable
24	**CMD**	CMD	**Command line status** This bit reflects the command line value of MSDC bus.
23:16	**DAT**	DAT	**Data line status** This bit reflects the data line value of MSDC bus. (8-bits)
15:12	**CDDBCE**	CDDEBOUNCE	**Card detection de-bounce timer** The register field specifies the time interval for card detection de-bounce. Its default value is 0. It means that de-bounce interval is one 32KHz cycle. The interval will extend one cycle time of 32KHz by increasing the counter by 1
1	**CDSTS**	CDSTS	**Card detection status** 1'b1: Card detection pin status is logic high 1'b1: Card detection pin status is logic high
0	**CDEN**	CDEN	**Card detect enable** The register bit is used to control the card detection circuit 1'b0: Card detection is disable 1'b1: Card detection is enable

1124000C MSDC_INT MSDC Interrupt Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										AC53FAIL	AC53DONE	GEAROUTBOUND	DMAPROTECT	GPDCSERR	BDCSERR	AC19DONE
Type										W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SDDCRCERR	SDDTO	DMAXFDNE	SDXFCPL	SDCSTS	SDRCRCER	SDCTO	SDCRDY	SDIOIRQ	DMAQEPTY	SDACDRCRCER	SDACDCTO	SDACDCRDY		MSDCCDSC	MMCIRQ
Type	W1C	W1C	W1C	W1C	RU	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C		W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**AC53FAIL**	AUTOCMD53_FAIL	**SD Auto command 53 status register**

Bit(s)	Mnemonic	Name	Description
			When auto-command 53 tuning sequence is failure , this register will be set to 1 by H/W.
21	**AC53DONE**	AUTOCMD53_DONE	**SD Auto command 53 status register** When auto-command 53 tuning sequence is finished , this register will be set to 1 by H/W.
20	**GEAROUTBOUND**	GEAR_OUT_BOUND	**the gear setting of delayline is out of boundary during SDIO Autocmd 53 On-Line tuning process**
19	**DMAPROTECT**	DMA_PROTECT	**there is write operation to DMA start address, length, start bit or last buf bit**
18	**GPDCSERR**	GPD_CS_ERR	**GPD checksum error detected**
17	**BDCSERR**	BD_CS_ERR	**BD checksum error detected**
16	**AC19DONE**	AUTOCMD19_DONE	**SD Auto command 19 status register** When auto-command 19 is enabled, H/W will clear this register to 0. As the tuning sequence is finished (32 times), this register will be set to 1 by H/W. S/W should check the AUTOCMD_STS0 and SUTOCMD_STS1 only when this bit is ON.
15	**SDDCRCERR**	SD_DATA_CRCERR	**SD Data CRC error interrupt** Indicates that MS/SD controller detects a CRC error after reading a block of data from the DAT line or SD/MMC signals a CRC error after writing a block of data to the DAT line. 1'b0: Otherwise 1'b1: MS/SD controller detected a CRC error after reading a block of data from the DAT line or SD/MMC signaled a CRC error after writing a block of data to the DAT line
14	**SDDTO**	SD_DATTO	**SD Data timeout interrupt** Indicates that SD/MMC controller detects a timeout condition while waiting for data token on the DAT line. This bit is for both data read and data write. For SD data read, timeout will occur when the read data is not presented. For SD data write, timeout will occur when the write data CRC status is not presented if PATCH_BIT[30] DETECT_WR_CRC_TIMEOUT = 1 1'b0: Otherwise 1'b1: SD/MMC controller detects a timeout condition while waiting for data token on the DAT line
13	**DMAXFDNE**	DMA_XFER_DONE	**DMA transfer done interrupt** The register bit indicates the status of data block transfer. 1'b0: Otherwise 1'b1: A data block was successfully transferred
12	**SDXFCPL**	SD_XFER_COMPLETE	**SD Data transfer complete interrupt** This bit indicates the transaction which contains data has completed. While performing tuning procedure (Execute Tuning is set to 1), SD_XFER_COMPLETE is not set to 1.
11	**SDCSTS**	SD_CSTS	**SD CSTA update interrupt** The register bit indicates any bit in the register SDC_CSTA is active, the register bit will be set to 1. S/W should clear the SDC_CSTA and this bit will be de-asserted automatically. 1'b0: No SD Memory Card interrupt 1'b1: SD Memory Card interrupt exists
10	**SDRCRCER**	SD_RESP_CRCERR	**SD Command CRC error interrupt** Indicates that SD/MMC controller detected a CRC error after reading a response from the CMD line. 1'b0: Otherwise

Bit(s)	Mnemonic	Name	Description
			1'b1: SD/MMC controller detected a CRC error after reading a response from the CMD line
9	**SDCTO**	SD_CMDTO	**SD Command timeout interrupt** Indicates that SD/MMC controller detected a timeout condition while waiting for a response on the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a timeout condition while waiting for a response on the CMD line
8	**SDCRDY**	SD_CMDRDY	**SD Command ready interrupt** For the command without response, the register bit will be 1 once the command completes on SD/MMC bus. For command with response without busy, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error. For command with response with busy in DAT0, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error and the DAT0 transited from busy to idle. 1'b0: Otherwise 1'b1: Command finish successfully without a CRC error
7	**SDIOIRQ**	SD_SDIOIRQ	**SD SDIO interrupt** This bit indicates the interrupt is sensed in the SDIO bus. 1'b0: No interrupt on SDIO bus 1'b1: Interrupt on SDIO bus
6	**DMAQEPTY**	DMA_Q_EMPTY	**DMA queue empty interrupt** This bit is used to indicate the current DMA queue is empty. Only for Descriptor mode and Enhance mode.
5	**SDACDRCRCER**	SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt** This bit is set when detecting a CRC error in the Auto command response.
4	**SDACDCTO**	SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt** This bit is set if no response is returned within a specified cycles(64T in spec) from the end bit of Auto command.
3	**SDACDCRDY**	SD_AUTOCMD_CMDRDY	**SD auto command ready interrupt** This bit is set if auto command is executed without CRC error or time out.
1	**MSDCCDSC**	MSDC_CDSC	**MSDC Card detection status change interrupt** The register bit indicates if any interrupt for memory card insertion/removal exists. Whenever memory card is inserted or removed and card detection circuit is enabled, i.e., the register bit CDEN in the register MSDC_PS is set to 1, the register bit will be set to 1. It will be reset when the register is read. 1'b0: Otherwise 1'b1: Card is inserted or removed
0	**MMCIRQ**	MMC_IRQ	**MMC card interrupt** 1'b0: Otherwise 1'b1: indicates that MMC card interrupt event occurs

11240010 MSDC_INTEN MSDC Interrupt Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										ENAC53FAIL	ENAC53DONE	ENGEAROUTBOU	ENDMAPROTECT	ENGPDCSERR	ENBDCSERR	ENAC19DONE

												ND				
Type										RW	RW	RW	RW	RW	RW	RW
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENSDDCRCERR	ENSDDTO	ENDMAXFDNE	ENSDXFCPL	ENSDCSTA	ENSDRCRCER	ENSDCTO	ENSDCRDY	ENSDIOIRQ	ENDMAQEPTY	ENSDACDRCRCER	ENSDACDCTO	ENSDACDCRDY		ENMSDCCDSC	ENMMCIRQ
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**ENAC53FAIL**	EN_AUTOCMD53_FAIL	**Auto-command 53 failure interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
21	**ENAC53DONE**	EN_AUTOCMD53_DONE	**Auto-command 53 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
20	**ENGEAROUTBOUND**	EN_GEAR_OUT_BOUND	**Gear out boundary interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
19	**ENDMAPROTECT**	EN_DMA_PROTECT	**DMA protection interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
18	**ENGPDCSERR**	EN_GPD_CS_ERR	**GPD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
17	**ENBDCSERR**	EN_BD_CS_ERR	**BD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
16	**ENAC19DONE**	EN_AUTOCMD19_DONE	**Auto-command 19 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
15	**ENSDDCRCERR**	EN_SD_DATA_CRCERR	**SD Data CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
14	**ENSDDTO**	EN_SD_DATTO	**SD Data timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
13	**ENDMAXFDNE**	EN_SD_DMA_XFER_DONE	**DMA transfer done interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
12	**ENSDXFCPL**	EN_SD_XFER_COMPLETE	**SD Data transfer complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
11	**ENSDCSTA**	EN_SD_CSTA	**SD CSTA update interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
10	**ENSDRCRCERR**	EN_SD_RESP_CRCERR	**SD Command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
9	**ENSDCTO**	EN_SD_CMDTO	**SD Command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

Bit(s)	Mnemonic	Name	Description
8	**ENSDCRDY**	EN_SD_CMDRDY	**SD Command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
7	**ENSDIOIRQ**	EN_SD_SDIOIRQ	**SD SDIO interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
6	**ENDMAQEPTY**	EN_DMA_Q_EMPTY	**DMA queue empty interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
5	**ENSDACDRCRCER**	EN_SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
4	**ENSDACDCTO**	EN_SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
3	**ENSDACDCRDY**	EN_AUTOCMD_CMDRDY	**SD auto command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
1	**ENMSDCCDSC**	EN_MSDC_CDSC	**MSDC Card detection status change interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
0	**ENMMCIRQ**	EN_MMC_IRQ	**MMC card interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

11240014 MSDC_FIFOCS MSDC FIFO Control and Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFOCLR								TXFIFOCNT							
Type	A0								RU							
Reset	0								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXFIFOCNT							
Type									RU							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**FIFOCLR**	FIFOCLR	**Embedded FIFO clear** Write this bit to 1 makes FIFO cleared. It will goes to 0 when FIFO is cleared. S/W needs to check this bit to make sure clearing FIFO sequence is done. This bit can be used when the data read/write sequence has error and need to clean the H/W FIFO.
23:16	**TXFIFOCNT**	TXFIFOCNT	**TX FIFO count for MSDC write** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved
7:0	**RXFIFOCNT**	RXFIFOCNT	**RX FIFO count for MSDC read** 8'd0: No data in FIFO

Bit(s)	Mnemonic	Name	Description
			8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved

11240018 MSDC_TXDATA MSDC TX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIOTXDATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIOTXDATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIOTXDATA**	PIO_TXDATA	**PIO mode TXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

1124001C MSDC_RXDATA MSDC RX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIORXDATA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIORXDATA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIORXDATA**	PIO_RXDATA	**PIO mode RXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

11240030 SDC_CFG SD Configuration Register 00100000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DTOC										INTBGP	SDIODE	SDIO		BUSWD	
Type	RW										RW	RW	RW		RW	
Reset	0	0	0	0	0	0	0	0			0	1	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ENWKUPINS	ENWKUPSDIOINT
Type															RW	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31:24	**DTOC**	DTOC	**Data Timeout Counter** The period from the end of the initial host read command or the last read data block in a multiple block read operation to the start bit of the next read data block requires at least two serial clock cycles. The counter is used to extend the period (Read Data Access Time) in unit of 1048576 serial clocks. 8'b00000000: Extend 1048576 more serial clock cycle 8'b00000001: Extend 1048576x2 more serial clock cycle 8'b00000010: Extend 1048576x3 more serial clock cycle 8'b11111111: Extend 1048576x 256 more serial clock cycle
21	**INTBGP**	INT_AT_BLOCK_GAP	**Interrupt at block Gap** This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. Setting to 0 disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the Host Driver detects an SD card insertion, it shall set this bit according to the CCCR of the SDIO card. 1'b0: Disables interrupt detection at the block gap 1'b1: Enables interrupt detection at the block gap
20	**SDIOIDE**	SDIO_INT_DET_EN	**SDIO interrupt detection enable** This bit is to inform the SD controller to sense the SDIO interrupt 1'b0: SDIO interrupt detection is disabled 1'b1: SDIO interrupt detection is enabled if the SDIO bit is also on
19	**SDIO**	SDIO	**SDIO mode enable bit** This bit is to enable the support to sense the SDIO interrupt and disable the R4 response CRC check for SDIO card 1'b0: SDIO mode is disabled 1'b1: SDIO mode is enabled
17:16	**BUSWD**	BUSWIDTH	**Bus width configuration** This field is used to define the SD/MMC bus width 2'b00: 1 bit mode 2'b01: 4 bit mode 2'b10: 8 bit mode 2'b11: reserved
1	**ENWKUPINS**	WAKEUP_INS_EN	**Card status change wakeup event enable bit** 1'b0: Disable wakeup event for card status change 1'b1: Enable wakeup event for card status change
0	**ENWKUPSDIOINT**	WAKEUP_SDIOINT_EN	**SDIO card interrupt wakeup event enable bit** 1'b0: Disable wakeup event for SDIO card interrupt 1'b1: Enable wakeup event for SDIO card interrupt

11240034 SDC_CMD SD Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD53	VOLSWTH	ACMD		LEN											
Type	RW	RW	RW		RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOIRQ	STOP	RW	DTYPE			RSPTYP			BREAK	CMD					
Type	RW	RW	RW	RW			RW			RW	RW					
Reset	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**ACMD53**	AUTO_CMD53	**Auto command 53 enable** This field should use exclusively with AUTO_CMD field. means that AUTO_CMD != 2'b00, this field should assign to 1'b0, if this filed is assigned to 1'b1, AUTO_CMD should set to 2'b00 1'b0: Disable Auto Command 53 1'b1: Enable Auto Command 53
30	**VOLSWTH**	VOL_SWTH	**Voltage switch command** 1'b0: Disable voltage switch detection 1'b1: Enable voltage switch detection
29:28	**ACMD**	AUTO_CMD	**Auto command enable** This field determines use of auto command functions. This function can be used in all modes including PIO/Basic DMA/Descriptor DMA/Enhanced Mode. There are two methods to stop Multiple-block read and write operation. (1) Auto CMD12 Enable Multiple-block read and write commands for memory require CMD12 to stop the operation. When ACMD-12 is used, MSDC issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the MSDC_INT register. The Host Driver shall not set this bit if the command does not require CMD12. In particular, secure commands defined in the Part 3 File Security specification do not require CMD12. (2) Auto CMD23 Enable When ACMD-23 is used, MSDC issues a CMD23 automatically before issuing a command specified in the CMD field. The Host Controller Version 3.00 and later shall support this function. By writing the Command register, MSDC issues a CMD23 first and then issues a command specified by the CMD field in SDC_CMD register. If response errors of CMD23 are detected, the second command is not issued. A CMD23 error is indicated in the MSDC_INT register. 32-bit block count value for CMD23 is set to SDC_BLOCK_NUM register. 2'b00: Disable Auto Command 2'b01: Enable Auto CMD12 2'b10: Enable Auto CMD23 2'b11: Enable Re-tuning CMD19
27:16	**LEN**	LEN	**Length** The register field is used to define the length of one block in unit of byte in a data transaction of block mode or the data length in unit of byte in data transaction of byte mode. The maximal value of block length is 2048 bytes. 12'b000000000000: Reserved 12'b000000000001: Block length is 1 byte 12'b000000000010: Block length is 2 byte 12'b011111111111: Block length is 2047 byte 12'b100000000000: Block length is 2048 byte
15	**GOIRQ**	GO_IRQ	**GO_IRQ command** The register bit indicates if the command is GO_IRQ_STATE (CMD40) and used only for MMC protocol. If the command is GO_IRQ_STATE, the period between command token and response token will not be limited. 1'b0: The command is not GO_IRQ_STATE 1'b1: The command is GO_IRQ_STATE
14	**STOP**	STOP	**Stop command** The register bit indicates if the command is a stop transmission command. It should be set to 1 when CMD12 (SD/MMC) or CMD52

Bit(s)	Mnemonic	Name	Description
			with I/O abort (SDIO) is to be issued. 1'b0: The command is not a stop transmission command 1'b1: The command is a stop transmission command
13	**RW**	RW	**Command read write selection** The register bit defines the command is a read command or write command. The register bit is valid only when the command will cause a transaction with data token. 1'b0: The command is a read command 1'b1: The command is a write command
12:11	**DTYPE**	DTYPE	**Data block selection** The register field defines data token type for the command. 2'b00: No data token for the command 2'b01: Single block transaction (only available in block mode) 2'b10: Multiple block transaction. (only available in block mode) 2'b11: Stream operation. It only shall be used in MMC protocol. (only available in block mode)
9:7	**RSPTYP**	RSPTYP	**Command response type** 3'b000: This command has no response. 3'b001: The command has R1/R5/R6/R7 response. The response token is 48-bit with CRC check (For SD/MMC/SDIO) (Not include the SDIO abort command) 3'b010: The command has R2 response. The response token is 136-bit (For SD/MMC) 3'b011: The command has R3 response. The response token is 48-bit response, no CRC check (For SD/MMC) 3'b100: The command has R4 response. The response token is 48-bit without CRC check (For SDIO) The response token is 48-bit with CRC check (For MMC) 3'b111: The command has R1b response. The response token is 48-bit (For SD/MMC/SDIO)
6	**BREAK**	BREAK	**Abort a pending MMC GO_IRQ command** It is only valid for a pending GO_IRQ_MODE command waiting for MMC interrupt response. 1'b0: Not a beak command 1'b1: Break a pending MMC GO_IRQ_MODE command in the controller.
5:0	**CMD**	CMD	**SD Memory Card command**

11240038 SDC_ARG SD Argument Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARG[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ARG[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ARG**	ARG	**Memory card controller argument register**

1124003C SDC_STS SD Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMCSWRCPL															CMD_WR_BUSY
Type	RU															W1C
Reset	0															0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															CMDBSY	SDCBSY
Type															RU	RU
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31	**MMCSWRCPL**	MMC_STREAM_WR_COMPL	**MMC Stream mode write data is all flushed to MMC card** S/W can use this bit to confirm last write data are flushed to MMC then issue STOP command. This bit is only valid when the command SDC_CMD.DTYPE=2'b11. 1'b0: Last Data are partially inside MSDC 1'b1: Last data are flushed to MMC card
16		CMD_WR_BUSY	
1	**CMDBSY**	CMDBUSY	**SD Command line busy status** S/W should always read this bit to make sure the command line is not busy before sending the next command. If the command is R1B or data read/write command, S/W should check SDCBUSY bit too. Note: When Auto command 12 is enabled, this bit will be asserted immediately after SDC_CMD is written and de-asserted after auto-command 12 finishes. 1'b0: No transmission is going on CMD line on SD bus 1'b1: There exists transmission going on CMD line on SD bus
0	**SDCBSY**	SDCBUSY	**SD controller busy status** 1'b0: SD controller is idle 1'b1: SD controller is busy

11240040 SDC_RESP0 SD Response Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP0**	RESP0	**Memory card controller response register 0**

11240044 SDC_RESP1 SD Response Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP1[31:16]															
Type	RU															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP1[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP1**	RESP1	**Memory card controller response register 1**

11240048 SDC_RESP2 SD Response Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP2[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP2[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP2**	RESP2	**Memory card controller response register 2**

1124004C SDC_RESP3 SD Response Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP3[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP3[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP3**	RESP3	**Memory card controller response register 3**

11240050 SDC_BLK_NUM SD Block Number Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BLKNUM[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BLKNUM[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0	**BLKNUM**	BLOCK_NUMBER	**Memory card controller Block number** This field indicates the block number of data transaction.

Bit(s)	Mnemonic	Name	Description
			32'd0: Reserved 32'd1: 1 data block 32'd2: 2 data block 32'd3: 3 data block 32'hffffffff: 4GB-1 data block

11240054 SDC_VOL_CHG SD Voltage Change Wait Time Register 00000145

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VCHGCNT															
Type	RW															
Reset	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15:0	**VCHGCNT**	VOL_CHG_WAIT_CNT	**This register define SD voltage change check wait time,wait time is clock frequency multiply VOL_WAIT_TIME**

11240058 SDC_CSTS SD Card Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS**	CSTS	**CSTS** The card status field in the response R1 or R1b field. Each bit can be write 1 clear individually.

1124005C SDC_CSTS_EN SD Card Status Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS_EN[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS_EN[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS_EN**	CSTS_EN	**CSTS_EN** This register is used to control which bit of the CSTA will generate

Bit(s)	Mnemonic	Name	Description
			the MSDC_INT.SDCSTA

11240060 **SDC_DATCRC_STS** **SD Card Data CRC Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCSSN								DCSSP							
Type	RU								RU							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8	**DCSSN**	DAT_CRCSTS_NEG	**MSDC DDR mode negative edge Read DATA CRC status** This register reflects the CRC status of data line[7:0] in DDR mode. The positive edge CRC status is shown in DAT_CRC_STS[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error
7:0	**DCSSP**	DAT_CRCSTS_POS	**MSDC read DATA CRC status** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error

11240080 **SD_ACMD_RESP** **SD ACMD Response Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMDRESP[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMDRESP[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMDRESP**	AUTOCMD_RESP	**SD Auto command response register** This register stores the response[39:8] of ACMD12/ACMD23/ACMD19.

11240084 **SD_ACMD19_TRG** **SD ACMD19 Target Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ACMDFTSEL			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**ACMDFTSEL**	FINE_TUNE_SEL	**SD Auto command 19 test target selection** After auto-command 19 is triggered, MSDC will only change the phase of the selected one and keep the value of other tuning registers. S/W can get the result from AUTOCMD_STS. There are total 32 phases for each delay line. 4'd0: Select PAD_CLK_TXDLY[5:0] as the target to run auto-command19. 4'd1: Select PAD_CMD_RXDLY[5:0] as the target to run auto-command19. 4'd2: Select PAD_DAT_RD_RXDLY[5:0] as the target to run auto-command19. 4'd3: Select PAD_DAT_WR_RXDLY[5:0] as the target to run auto-command19. 4'd4: Select DAT0_RD_DLY[5:0] as the target to run auto-command19. 4'd5: Select DAT1_RD_DLY[5:0] as the target to run auto-command19. 4'd6: Select DAT2_RD_DLY[5:0] as the target to run auto-command19. 4'd7: Select DAT3_RD_DLY[5:0] as the target to run auto-command19. 4'd8: Select DAT4_RD_DLY[5:0] as the target to run auto-command19. 4'd9: Select DAT5_RD_DLY[5:0] as the target to run auto-command19. 4'd10: Select DAT6_RD_DLY[5:0] as the target to run auto-command19. 4'd11: Select DAT7_RD_DLY[5:0] as the target to run auto-command19. 4'd12: Select CMD_RESP_RXDLY[5:0] as the target to run auto-command19. Others: Reserved

11240088 **SD_ACMD19_STS** **SD ACMD19 Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD19STS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMD19STS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMD19STS**	AUTOCMD19_STS	**SD Auto command 19 test result register** When auto-command 19 is enabled, H/W will automatically try 32 times of command-19 and store the result into this register. This register contains 1st to 32th results in bit[0:31]

1124008C DMA_SA_HIGH 4BIT DMA Current Address Resgiter of high 4bit 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													DMASAHIGH4BIT			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**DMASAHIGH4BIT**	DMA_SURR_ADDR_HIGH4BIT	**it is usend to set high 4bit address of start address because 64G dram need 36bit address**

11240090 DMA_SA DMA Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMASA[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMASA[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMASA**	DMA_STR_ADDR	**The start address of the DMA address** This register is used to set the start address of the DMA. In DMA basic mode, this field indicates the source or destination address of the data transfer which depends on the command. In descriptor base DMA, this is the descriptor chain start address.

11240094 DMA_CA DMA Current Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMACA**	DMA_CURR_ADDR	**The current address of the DMA address** This register is used to read the current address of the DMA descriptor chain.

11240098 DMA_CTRL DMA Control Register 00006008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		**BSTSZ**			**SPLIT1K**	**LASTBF**	**DMAALIGN**	**DMAMOD**					**READYM**	**DMARSM**	**DMASTOP**	**DMASTART**
Type		RW			RW	RW	RW	RW					RO	WO	A0	WO
Reset		1	1	0	0	0	0	0					1	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**BSTSZ**	BURST_SIZE	**DMA burst size** This field is used to specify the maximum transfer bytes allowed at the device per DMA burst. This field can not be modified when the DMA status is 1. 3'd3: 8 Bytes 3'd4: 16 Bytes 3'd5: 32 Bytes 3'd6: 64 Bytes Other: Reserved
11	**SPLIT1K**	DMA_SPLIT_1K	**This field is used to specify whether split burst when corss 1K boundry address** 1'b0: 1K boundary not split 1'b1: 1K boundary split
10	**LASTBF**	LAST_BUF	**Last buffer of the basic DMA mode** This field indicates the last buffer in the basic DMA mode
9	**DMAALIGN**	DMA_ALIGN	**This field is used to specify whether address aligment burst size** 1'b0: do not DAM burst size alignment 1'b1: DAM burst size alignment
8	**DMAMOD**	DMA_MODE	**DMA operation mode** This field indicates operation mode of DMA 1'b0: Basic DMA mode 1'b1: Descriptor base DMA mode
3	**READYM**	AHB_READYM	**only for debug when dma hang,sw can check if ahb bus is ok when gdma is hang** 1: bus is normal 0: bus not normal
2	**DMARSM**	DMA_RESUME	**DMA resume control register** This bit is used to resume the DMA transaction. Read always return 0
1	**DMASTOP**	DMA_STOP	**DMA Stop control register** This bit is used to stop the DMA transaction. When SW issue STOP command, SW must wait this bit de-assert or DMA inactive to guarantee stop done.
0	**DMASTART**	DMA_START	**DMA start control register** This bit is used to start the DMA transaction. Read always return 0

1124009C DMA_CFG DMA Configuration Register 00000004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															**OUTBOUND**	**DMACHKSU**

															STOP DMA	M12B
Type															RW	RW
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MSDCACTIVEEN				AHBHPROT2EN							LOCKDISABLE	DSCPCSEN	DMASTS
Type			RW				RW							RW	RW	RU
Reset			0	0			0	0						1	0	0

Bit(s)	Mnemonic	Name	Description
17	**OUTBOUNDSTOPDMA**	OUT_BOUND_STOP_DMA	**This register will determine whether stop Enhance DMA if gear setting is out-of-boundary during send training data** 1'b0: Enhance DMA will continue even if gear setting is out-of-boundary during send training data 1'b1: Enhance DMA will stop if gear setting is out-of boundary during send training data
16	**DMACHKSUM12B**	DMA_CHK_SUM_12B	**This register indicates GPD/BD checksum cover 16byte or 12byte** 1'b0: GPD/BD checksum cover 16byte 1'b1: GPD/BD checksum only cover 12byte
13:12	**MSDCACTIVEEN**	MSDC_ACTIVE_EN	**This register will indicate how to control msdc_active** 2'b00: dynamic control msdc_active 2'b01: msdc_active = 0 2'b10: msdc_active = 1 2'b11: Reserved
9:8	**AHBHPROT2EN**	AHB_HPROT_2_EN	**This register will determine how to control hprot_2 pin of AHB bus** AHB_HPROT_2_EN = 2'b00, and Basic DMA Mode All the write transfers of a burst will access by bufferable mode except the last burst of DMA AHB_HPROT2_2_EN=2'b00, and Descriptor DMA Mode all the write transfers of a burst will access by bufferable mode except HW own update transfer 2'b00: dynamic control hprot_2 2'b01: hprot_2 = 0 2'b10: hprot_2 = 1
2	**LOCKDISABLE**	LOCK_DISABLE	**should disable lock in order to improve emi efficient** 1'b0: enable ahb lock 1'b1: disable ahb lock
1	**DSCPCSEN**	DMA_DSCP_CS_EN	**DMA descriptor checksum enable** This bit is used to enable or disable the descriptor checksum validation function for the descriptor. This field can not be modified when the DMA status is 1.
0	**DMASTS**	DMA_STATUS	**DMA status** This bit is used to indicate the status of the DMA. 1'b0: DMA engine is inactive 1'b1: DMA engine is active

112400A0 SW_DBG_SEL MSDC S/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGSEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**SWDBGSEL**	DBG_SEL	**MSDC debug selection** This contain is reserved!

112400A4 SW_DBG_OUT MSDC S/W Debug Output Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SWDBGO[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGO[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SWDBGO**	DBG_OUT	**MSDC debug output** 32 bit output selected by SW_DBG_SEL register

112400A8 DMA_LENGTH DMA Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	XFSZ[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	XFSZ[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**XFSZ**	XFER_SIZE	**DMA total transfer size** This field is used to specify the number of DMA transfer byte required for the movement of source data through DMA. This field is only valid in basic DMA mode.

112400B0 PATCH_BIT0 MSDC Patch Bit Register 0 403C0006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PTCH 31	PTCH 30	PTCH 29	PTCH 28	PTCH 27	PTCH 26	PTCH22				PTCH18				PTCH 17	PTCH 16
Type	RW	RW	RW	RW	RW	RW	RW				RW				RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PTCH 15	CKGDLYS					INTCKS			DESC UP	PTCH 5	PTCH 4	PTCH 3	PTCH 02	PTCH 01	
Type	RW	RW					RW			RW	RW	RW	RW	RW	RW	

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	

Bit(s)	Mnemonic	Name	Description
31	**PTCH31**	EN_MMC_DRV_RESP	**Enable MSDC always drives bus when output wakeup response (BREAK)** 1'b0: Disable 1'b1: Enable
30	**PTCH30**	DETECT_WR_CRC_TIMEOUT	**MSDC write data CRC phase timeout detection** 1'b0: Not detect CRC phase timeout 1'b1: detect CRC phase timeout
29	**PTCH29**	SPC_ALWAYS_PUSH	**SPC Buffer push mechanism** 1'b0: Push the buffer only when read transfer is on-going 1'b1: Always push the buffer
28	**PTCH28**	SDIO_INT_DLY_SEL	**SDIO interrupt latch time selection** 1'b0: Latch the data line value in internal SDIO interrupt period 1'b1: Latch the data line value in 1 clock delay of internal SDIO interrupt period
27	**PTCH27**	SDC_CMD_CMDFAIL_SEL	**SDIO interrupt period recovery selection** 1'b0: SDIO interrupt period will re-start after a CMD12 or CMD52 command is issued 1'b1: SDIO interrupt period whenever DAT line is not busy
26	**PTCH26**	SDC_CMD_IDRT_SEL	**SD identification response time selection** The register bit indicates if the command has a response with NID (that is, 5 serial clock cycles as defined in SD Memory Card Specification Part 1 Physical Layer Specification version 1.0) response time. The register bit is valid only when the command has a response token. Thus the register bit must be set to 1 for CMD2 (ALL_SEND_CID) and ACMD41 (SD_APP_OP_CMD). 1'b0: Otherwise. 1'b1: The command has a response with NID response time.
25:22	**PTCH22**	SDC_CFG_WDOD	**SD Write Data Output Delay** The period from finish of the response for the initial host write command or the last write data block in a multiple block write operation to the start bit of the next write data block requires at least two serial clock cycles. The register field is used to extend the period (Write Data Output Delay) in unit of one serial clock. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
21:18	**PTCH18**	SDC_CFG_BSYDLY	**SD R1B busy detection mode** The register field is only valid for the commands with R1b response. If the command has a response of R1b type, MS/SD controller must monitor the data line 0 for card busy status from the bit time that is two serial clock cycles after the command end bit to check if operations in SD/MMC Memory Card have finished. The register field is used to expand the time between the command end bit and end of detection period to detect card busy status. If time is up and there is no card busy status on data line 0, then the controller will abandon the detection. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
17	**PTCH17**	SDIO_CFG_INTC_SEL	**SDIO Interrupt model selection** 1'b0: Only when data line [1] = 0 and then trigger SDIO interurpt event

Bit(s)	Mnemonic	Name	Description
			1'b1: Only when data line [3:0] = 4'b1101 and then trigger SDIO interurpt event
16	**PTCH16**	MSDC_BLKNUM_SEL	**Configuration support ACMD23 reliable/force prog etc. feature** 1'b0: Support ACMD23 reliable/force prog etc. feature 1'b1: Don't support ACMD23 reliable/force prog etc. feature
15	**PTCH15**	MSDC_FIFO_RD_DIS	**MSDC RXFIFO Read Disable** 1'b0: Disable FIFO read permission to RXFIFO in PIO mode 1'b1: Enable FIFO read permission to RXFIFO in PIO mode
14:10	**CKGDLYS**	CKGEN_MSDC_DLY_SEL	**CKBUF in CKGEN Delay Selection** Total 32 stages
9:7	**INTCKS**	INT_DAT_LATCH_CK_SEL	**Internal MSDC clock phase selection** Total 8 stages, each stage can delay 1 clock period of msdc_src_ck
6	**DESCUP**	DESC_UP_SEL	**sd transfer done int should be issue when GPD have been update** 1'b1: enable new function for generate sd transfer done int 1'b0: use old function for generate sd transfer done int
5	**PTCH5**	ACMD53_FAIL_ONE_SHOT	**determine interrupt method of AUTOCMD53_FAIL** 1'b0: AUTOCMD53_FAIL interrupt will assert whenever CMD/DAT crc error occur 1'b1: AUTOCMD53_FAIL interrupt will assert only when AUTOCMD53_DONE assert if there is CMD/DAT crc error
4	**PTCH4**	MASK_ACMD53_CRC_ERR_INTR	**mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence** 1'b0: enable CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence 1'b1: mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence
3	**PTCH3**	RD_DAT_SEL	**This field is used to define whther used rising or falling buf data for SDR mode** 1'b0: Used rising buf data for SDR mode 1'b1: Used falling buf data for SDR mode
2	**PTCH02**	DIS_REFLECT_CMDWR_WHEN_BSY	**Enable SD command register write montior** 1'b0: Enable monitor function 1'b1: Disable monitor function
1	**PTCH01**	EN_SDC_ODD_8BIT_SUP	**Enable SD odd number support for 8-bit data bus** 1'b0: Disable 1'b1: Enable

112400B4 **PATCH_BIT1** **MSDC Patch Bit Register 1** **FFB00009**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSHBFCKEN	MRCTLCKEN	MWCTLCKEN	MSDCKEN	MACMDCKEN	MVOLDTCKEN	MPSCCKEN	MSPCCKEN	HGDMACKEN		DCMEN	DCMDIVSEL1				SINGLEBURST
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW				RW
Reset	1	1	1	1	1	1	1	1	1		1	1				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESETGDMA		BIAS28R0	BIAS28R1	BIAS28R2				GETCRCMARGIN	GETBUSYMARGIN	CMDTA			WRTA		
Type	RW		RW	RW	RW				RW	RW	RW			RW		
Reset	0		0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
31	**MSHBFCKEN**	MSDC_CK_SHBFF_CKEN	**msdc_src_ck clock enable bit for SHBFF** 1'b0: Disable 1'b1: Enable
30	**MRCTLCKEN**	MSDC_CK_RCTL_CKEN	**msdc_src_ck clock enable bit for RCTL** 1'b0: Disable 1'b1: Enable
29	**MWCTLCKEN**	MSDC_CK_WCTL_CKEN	**msdc_src_ck clock enable bit for WCTL** 1'b0: Disable 1'b1: Enable
28	**MSDCKEN**	MSDC_CK_SD_CKEN	**msdc_src_ck clock enable bit for SD** 1'b0: Disable 1'b1: Enable
27	**MACMDCKEN**	MSDC_CK_ACMD_CKEN	**msdc_src_ck clock enable bit for ACMD** 1'b0: Disable 1'b1: Enable
26	**MVOLDTCKEN**	MSDC_CK_VOLDET_CKEN	**msdc_src_ck clock enable bit for VOLDET** 1'b0: Disable 1'b1: Enable
25	**MPSCCKEN**	MSDC_CK_PSC_CKEN	**msdc_src_ck clock enable bit for PSC** 1'b0: Disable 1'b1: Enable
24	**MSPCCKEN**	MSDC_CK_SPC_CKEN	**msdc_src_ck clock enable bit forSPC** 1'b0: Disable 1'b1: Enable
23	**HGDMACKEN**	AHB_CK_GDMA_CKEN	**hclk_ck clock enable bit for GDMA** 1'b0: Disable 1'b1: Enable
21	**DCMEN**	DCM_EN	**host enable dcm for low power,when there is no transfer the ahb_ck and msdc_src_ck will divder from spurce clk** 1'b1: disable 1'b0: enable
20	**DCMDIVSEL1**	DCM_DIV_SEL1	**for spec4.5 divider msdc_src_ck from source clk,for spec50 divider ahb_ck from source clk** 1'b1: 32 divder from source clk 1'b0: 16 divder from source clk
16	**SINGLEBURST**	ENABLE_SINGLE_BURST	**the ahb bus will not support incr1 burst type in future.And it will only affect AHB bus msdc design,not affect AXI bus design** 1'b0: hw will send incr1 burst type 1'b1: hw will send single burst typr instead of incr1 type
15	**RESETGDMA**	RESET_GDMA	**sw can sw reset gdma when design hang** 1 'b1: reset gdma 1'b0: not reset gdma
13	**BIAS28R0**	BIAS_EXTBIAS_28NM	**28NM BIAS Controler register 0**
12	**BIAS28R1**	BIAS_EN18IO_28NM	**28NM BIAS Controler register 1**
11:8	**BIAS28R2**	BIAS_TUNE_28NM	**28NM BIAS Controler register 2**
7	**GETCRCMARGIN**	GET_CRC_MARGIN	**it will add margin for get crc status when card resp crc not match spec 2cycle from endbit** 1'b0: 8 cycle reserved for get crc status from write data crc endbit 1'b1: 16 cycle reserved for get crc status from write data crc endbit
6	**GETBUSYM**	GET_BUSY_MARG	**it will add margin for get busy state of data0**

Bit(s)	Mnemonic	Name	Description
	ARGIN	IN	1'b0: 1 cycle reserved for get busy state from src status endbit 1'b1: 3cycle reserved for get busy state from src status endbit
5:3	**CMDTA**	CMD_RSP_TA_CNTR	**CMD response turn around period** The turn around cycle = CMD_RSP_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode, this register should be set to 0
2:0	**WRTA**	WRDAT_CRCS_TA_CNTR	**Write data and CRC status turn around period** The turn around cycle = WRDAT_CRCS_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode,this register should be set to 0

112400B8 PATCH_BIT2 MSDC Patch Bit Register 2 14801803

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCSTSENSEL			CFGCRCSTS	CFGCRCSTSCNT		CFGCRCSTSEDGE	CFG_CRCSTS_SEL	POPENCNT					RESPSTSENSEL		
Type	RW			RW	RW		RW	RW	RW					RW		
Reset	0	0	0	1	0	1	0	0	1	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFGRESP	CFGRESPCNT			INTCRESPSEL		CFGRDAT	CFGRDATCNT					RESPWAITCNT		SUPPORT64G	ENHANCEGPD
Type	RW	RW			RW		RW	RW					RW		RW	RW
Reset	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:29	**CRCSTSENSEL**	CRCSTS_LATCH_EN_SEL	**This register used configuration latch CRC Status enable signal for async fifo in emmc45** 3'b000: latch CRC Status enable signal not delay 3'b001: latch CRC Status enable signal delay 1T msdc_ck 3'b010: latch CRC Status enable signal delay 2T msdc_ck 3'b011: latch CRC Status enable signal delay 3T msdc_ck 3'b111: latch CRC Status enable signal delay 7T msdc_ck
28	**CFGCRCSTS**	CFG_CRCSTS	**This register used configuration CRC Status path selection,this setting only used emmc4.5 feature** 1'b0: Latch CRC Status select delay-line path 1'b1: Latch CRC Status select async fifo path
27:26	**CFGCRCSTSCNT**	CFG_CRCSTS_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 2'b00: push 0 data in async fifo when start pop out data from async fifo 2'b01: push 1 data in async fifo when start pop out data from async fifo 2'b10: push 2 data in async fifo when start pop out data from async fifo 2'b11: push 3 data in async fifo when start pop out data from async fifo
25	**CFGCRCSTSEDGE**	CFG_CRCSTS_EDGE	**This register configuration used rising async fifo or falling async fifo** 1'b0: async fifo latch CRC Status used rising async fifo 1'b1: async fifo latch CRC Status used falling async fifo

Bit(s)	Mnemonic	Name	Description
24		CFG_CRCSTS_SEL	**This register configuration async fifo path selection** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
23:20	**POPENCNT**	POP_EN_CNT	**pop enable counter** This field is used to define how many write pointer and the read pointer margin began to pop data transfer
18:16	**RESPSTSENSEL**	RESP_LATCH_EN_SEL	**This register used configuration latch CMD Response enable signal for async fifo in emmc45** 3'b000: latch CMD Response enable signal not delay 3'b001: latch CMD Response enable signal delay 1T msdc_ck 3'b010: latch CMD Response enable signal delay 2T msdc_ck 3'b011: latch CMD Response enable signal delay 3T msdc_ck 3'b111: latch CMD Response enable signal delay 7T msdc_ck
15	**CFGRESP**	CFG_RESP	**This register used configuration CMD Response path selection,this setting only used emmc4.5 feature** 1'b0: Latch CMD Response select async fifo path 1'b1: Latch CMD Response select delay-line path
14:12	**CFGRESPCNT**	CFG_RESP_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 3'b000: push 0 data in async fifo when start pop out data from async fifo 3'b001: push 1 data in async fifo when start pop out data from async fifo 3'b111: push 7 data in async fifo when start pop out data from async fifo
11	**INTCRESPSEL**	INTC_RESP_SEL	**This register configuration BREAK command async fifo path** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
9	**CFGRDAT**	CFG_RDAT	**This register used configuration read data path** 1'b0: read data path by pass delay line 1'b1: read data path through delay line
8:4	**CFGRDATCNT**	CFG_RDAT_CNT	**This register used configuration read data path delay line**
3:2	**RESPWAITCNT**	RESP_WAIT_CNT	**This register used configuration cmd response timeout,timeout cycle is 65T+16*RESP_WAIT_CNT** 2'b00: cmd response timeout is 65T 2'b01: cmd response timeout is 65T+ 16*1T 2'b01: cmd response timeout is 65T+ 16*2T 2'b01: cmd response timeout is 65T+ 16*3T
1	**SUPPORT64G**	SUPPORT_64G	**This register used which proj support high 64G dram space access** 1'b1: support 64G dram access 1'b0: not support 64G dram access
0	**ENHANCEGPD**	ENHANCE_WAIT_GPD	**if sw clear int when gpd not update finish,design will hang.so you can set this bt to 1 to avoid this iuuse in enhance write mode** 1'b1: use new HW code for update gpd in enhance mode 1'b0: use old HW code

112400C0 DAT0_TUNE_C DAT0 Tune Result Register 00000000

RC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT0CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT0CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	DAT0CRCSTS	DAT0_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT0 into this register. This register contains 1st to 32th results in bit[0:31]

112400C4 DAT1_TUNE_CRC DAT1 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT1CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT1CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	DAT1CRCSTS	DAT1_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT1 into this register. This register contains 1st to 32th results in bit[0:31]

112400C8 DAT2_TUNE_CRC DAT2 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT2CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT2CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	DAT2CRCSTS	DAT2_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT2 into this register.

Bit(s)	Mnemonic	Name	Description
			This register contains 1st to 32th results in bit[0:31]

112400CC **DAT3_TUNE_CRC** **DAT3 Tune Result Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT3CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT3CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT3CRCSTS**	DAT3_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT3 into this register. This register contains 1st to 32th results in bit[0:31]

112400D0 **CMD_TUNE_CRC** **CMD Tune Result Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CMDCRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMDCRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CMDCRCSTS**	CMD_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of CMD into this register. This register contains 1st to 32th results in bit[0:31]

112400D4 **SDIO_TUNE_WIND** **SDIO Tune Window Register 0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												TUNEWINDOW				
Type												RW				
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:0	**TUNEWINDOW**	TUNE_WINDOW	**SDIO Auto command 53 tuning window setting. the window range will be** 1. for CMD PAD low bound: max(0, PAD_CMD_RXDLY-TUNE_WINDOW) high bound: min(31, PAD_CMD_RXDLY+TUNE_WINDOW) 2. for DAT PAD low bound: max(0, DAT(0:3)_RD_DLY-TUNE_WINDOW) high bound:min(31: DAT(0:3)_RD_DLY+TUNE_WINDOW)

112400F0 PAD_TUNE0 MSDC Pad Tuning Register0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKTDLY					CMDRRDLY					CMDRRDLYSEL	CMDRDLY				
Type	RW					RW					RW	RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDLYSEL		DATRRDLYSEL	DATRRDLY					DELAYEN			DATWRDLY				
Type	RW		RW	RW					RW			RW				
Reset	0		0	0	0	0	0	0	0			0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:27	**CLKTDLY**	PAD_CLK_TXDLY	**CLK Pad TX Delay Control** This register is used to add delay to CLK phase. Total 32 stages
26:22	**CMDRRDLY**	PAD_CMD_RESP_RXDLY	**CMD Response Internal Delay Line Control** This register is used to fine-tune response phase latched by MSDC internal clock Total 32 stages
21	**CMDRRDLYSEL**	PAD_CMD_RD_RXDLY_SEL	**Decide CMD Response pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY**	PAD_CMD_RXDLY	**CMD Pad RX Delay Line1 Control** This register is used to fine-tune CMD pad macro respose latch timing Total 32 stages
15	**RXDLYSEL**	PAD_RXDLY_SEL	**Decide rx delay line tune data path or clock path** 1'b0: rx delay line tune data path 1'b1: rx delay line tune clock path
13	**DATRRDLYSEL**	PAD_DAT_RD_RXDLY_SEL	**Decide rx data pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY**	PAD_DAT_RD_RXDLY	**DAT Pad RX Delay Line1 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages
7	**DELAYEN**	DELAY_EN	**enable all delay cell toogle when power on** 1'b0: disable delay cell toogle default 1'b1: enable delay cell toogle default

Bit(s)	Mnemonic	Name	Description
4:0	**DATWRDLY**	PAD_DAT_WR_RXDLY	**Write Data Status Internal Delay Line Control** This register is used to fine-tune write status phase latched by MSDC internal clock Total 32 stages

112400F4 PAD_TUNE1 MSDC Pad Tuning Register1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											CMDRRDLY2SEL	CMDRDLY2				
Type											RW	RW				
Reset											0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DATRRDLY2SEL	DATRRDLY2												
Type			RW	RW												
Reset			0	0	0	0	0	0								

Bit(s)	Mnemonic	Name	Description
21	**CMDRRDLY2SEL**	PAD_CMD_RD_RXDLY2_SEL	**Decide CMD Response pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY2**	PAD_CMD_RXDLY2	**CMD Pad RX Delay Line2 Control** This register is used to fine-tune CMD pad macro respose latch timing in data path Total 32 stages
13	**DATRRDLY2SEL**	PAD_DAT_RD_RXDLY2_SEL	**Decide rx data pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY2**	PAD_DAT_RD_RXDLY2	**DAT Pad RX Delay Line2 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages

112400F8 DAT_RD_DLY0 MSDC Data Delay Line Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY								DAT1RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY								DAT3RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY**	DAT0_RD_DLY	**DAT0 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY**	DAT1_RD_DLY	**DAT1 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

Bit(s)	Mnemonic	Name	Description
12:8	**DAT2RDDLY**	DAT2_RD_DLY	**DAT2 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY**	DAT3_RD_DLY	**DAT3 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

112400FC DAT_RD_DLY1 MSDC Data Delay Line Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY								DAT5RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY								DAT7RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY**	DAT4_RD_DLY	**DAT4 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY**	DAT5_RD_DLY	**DAT5 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY**	DAT6_RD_DLY	**DAT6 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY**	DAT7_RD_DLY	**DAT7 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

11240100 DAT_RD_DLY2 MSDC Data Delay Line Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY2								DAT1RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY2								DAT3RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY2**	DAT0_RD_DLY2	**DAT0 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY2**	DAT1_RD_DLY2	**DAT1 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT2RDDLY2**	DAT2_RD_DLY2	**DAT2 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY2**	DAT3_RD_DLY2	**DAT3 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11240104 DAT_RD_DLY3 MSDC Data Delay Line Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY2								DAT5RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY2								DAT7RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY2**	DAT4_RD_DLY2	**DAT4 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY2**	DAT5_RD_DLY2	**DAT5 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY2**	DAT6_RD_DLY2	**DAT6 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY2**	DAT7_RD_DLY2	**DAT7 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11240110 HW_DBG_SEL MSDC H/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBGWSEL	DBG3SEL						DBG2SEL							
Type		RW	RW						RW							
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DBG1SEL						DBG0SEL							
Type			RW						RW							
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30	**DBGWSEL**	HW_DBG_WRAP_SEL	**H/W debug output selection for wrapper** 0: Select original debug pins 1: Select wrapper debug pins
29:24	**DBG3SEL**	HW_DBG3_SEL	**H/W debug output selection**
23:16	**DBG2SEL**	HW_DBG2_SEL	**H/W debug output selection**
13:8	**DBG1SEL**	HW_DBG1_SEL	**H/W debug output selection**
7:0	**DBG0SEL**	HW_DBG0_SEL	**H/W debug output selection**

11240114 MAIN_VER MSDC Main Version Register 20140512

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINVER[31:16]															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAINVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:0	**MAINVER**	MAIN_VER	**Main Version**

11240118 ECO_VER MSDC ECO Version Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECOVER[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECOVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ECOVER**	ECO_VER	**ECO Version**

Module name: MSDC Base address: (+0h)

Address	Name	Width	Register Function
11250000	**MSDC_CFG**	32	**MSDC Configuration Register** The register is for general configuration of the MS/SD controller.
11250004	**MSDC_IOCON**	32	**MSDC IO Configuration Register** The register contains the receiver path data latch timing control and interface control bits.
11250008	**MSDC_PS**	32	**MSDC Pin Status Register** The register is used to storing card detection and write protection pin status. Card detection status can be disabled.
1125000C	**MSDC_INT**	32	**MSDC Interrupt Register** The register contains the status of interrupts. Note that the register still shows the status of interrupt even though the interrupt is disabled.
11250010	**MSDC_INTEN**	32	**MSDC Interrupt Enable Register** The register contains the related enable bit of interrupts.
11250014	**MSDC_FIFOCS**	32	**MSDC FIFO Control and Status Register** The register contains the control and status of embedded 128B FIFO.
11250018	**MSDC_TXDATA**	32	**MSDC TX Data Port Register** The register is for PIO mode only. Used to input MSDC write data to card. The access can be AHB 1B/2B/4B
1125001C	**MSDC_RXDATA**	32	**MSDC RX Data Port Register** The register is for PIO mode only. Used to read back MSDC read data from card. The access can be AHB 1B/2B/4B.
11250030	**SDC_CFG**	32	**SD Configuration Register** The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
11250034	**SDC_CMD**	32	**SD Command Register** The register defines a SD Memory Card command and its attributes. Before MS/SD controller issues a transaction onto SD

			bus, application shall specify other relative settings such as argument for command. After writing the register by the application, MS/SD controller will issue the corresponding transaction onto SD serial bus. If the command is GO_IDLE_STATE, the controller will have serial clock on SD/MMC bus run 128 cycles before issuing the command.
11250038	**SDC_ARG**	32	**SD Argument Register** The register contains the argument of the SD/MMC Memory Card command.
1125003C	**SDC_STS**	32	**SD Status Register** The register reflects SD bus status and contains MMC stream write status.
11250040	**SDC_RESP0**	32	**SD Response Register 0** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11250044	**SDC_RESP1**	32	**SD Response Register 1** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
11250048	**SDC_RESP2**	32	**SD Response Register 2** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
1125004C	**SDC_RESP3**	32	**SD Response Register 3** The register contains parts of the last SD/MMC Memory Card bus response. The register fields SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3 are composed of the last SD/MMC Memory card bus response. For response of type R2, that is, response of the command ALL_SEND_CID, SEND_CSD and SEND_CID, only bit 127 to 0 of response token is stored in the register field SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3. SDC_RESP0 = bit 31~0 SDC_RESP1 = bit 63~32 SDC_RESP2 = bit 95~64 SDC_RESP3 = bit 127~96 For response of type R1b in auto CMD12 or R1 in auto CMD23, bit 39 to 8 of response token is stored in the register field of SDC_RESP3. For the responses of other types, only bit 39 to 8 of response token is stored in the register field SDC_RESP0.
11250050	**SDC_BLK_NUM**	32	**SD Block Number Register** This register defines the block number for the block transaction. For single read/write, this register should be set to 1. For multiple read/write, this register should be set to larger than 1. Set to 0 will cause unexpected result.
11250054	**SDC_VOL_CHG**	32	**SD Voltage Change Wait Time Register** This register define SD voltage change check wait time
11250058	**SDC_CSTS**	32	**SD Card Status Register** After commands with R1 and R1b response, this register will contain the status of the SD/MMC card
1125005C	**SDC_CSTS_EN**	32	**SD Card Status Enable Register** This register is used to control which bit of the SDC_CSTS will generate the MSDC_INT.SD_CSTA interrupt.
11250060	**SDC_DATCRC_STS**	32	**SD Card Data CRC Status Register** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read
11250080	**SD_ACMD_RESP**	32	**SD ACMD Response Register** This register stores the response of auto command from SD card
11250084	**SD_ACMD19_TRG**	32	**SD ACMD19 Target Register** This register is used to select target delay line to run ACMD19 sequence.

11250088	SD_ACMD19_STS	32	**SD ACMD19 Status Register** This register stores the result of auto command 19 from SD card
1125008C	DMA_SA_HIGH4BIT	32	**DMA Current Address Resgiter of high 4bit** This register contain the start address high 4bit of 36bit address for 64G dram access
11250090	DMA_SA	32	**DMA Start Address Register** This register contains the start address of the DMA descriptor
11250094	DMA_CA	32	**DMA Current Address Register** This register contains the current DMA address
11250098	DMA_CTRL	32	**DMA Control Register** This register is used to control the DMA operation.
1125009C	DMA_CFG	32	**DMA Configuration Register** This register is used to configure the DMA operation.
112500A0	SW_DBG_SEL	32	**MSDC S/W Debug Selection Register** This register is used to select S/W debug output
112500A4	SW_DBG_OUT	32	**MSDC S/W Debug Output Register** This register shows the selected debug output
112500A8	DMA_LENGTH	32	**DMA Length Register** This register is used to set Basic DMA operation length
112500B0	PATCH_BIT0	32	**MSDC Patch Bit Register 0** This register can configure the patch function. For normal function, these bit should keep in default value
112500B4	PATCH_BIT1	32	**MSDC Patch Bit Register 1** This register can configure the patch function. For normal function, these bit should keep in default value
112500B8	PATCH_BIT2	32	**MSDC Patch Bit Register 2** This register can configure the patch function. For normal function, these bit should keep in default value
112500C0	DAT0_TUNE_CRC	32	**DAT0 Tune Result Register** This register record on-line tuning result for DAT0 line
112500C4	DAT1_TUNE_CRC	32	**DAT1 Tune Result Register** This register record on-line tuning result for DAT1 line
112500C8	DAT2_TUNE_CRC	32	**DAT2 Tune Result Register** This register record on-line tuning result for DAT2 line
112500CC	DAT3_TUNE_CRC	32	**DAT3 Tune Result Register** This register record on-line tuning result for DAT3 line
112500D0	CMD_TUNE_CRC	32	**CMD Tune Result Register** This register record on-line tuning result for CMD line
112500D4	SDIO_TUNE_WIND	32	**SDIO Tune Window Register 0** This register define tuning window size for SDIO on-line CRC tuning feature
112500F0	PAD_TUNE0	32	**MSDC Pad Tuning Register0** This register can configure the delay line embedded in Pad Macro
112500F4	PAD_TUNE1	32	**MSDC Pad Tuning Register1** This register can configure the delay line embedded in Pad Macro
112500F8	DAT_RD_DLY0	32	**MSDC Data Delay Line Register 0** This register can configure the delay line embedded in Pad Macro
112500FC	DAT_RD_DLY1	32	**MSDC Data Delay Line Register 1** This register can configure the delay line embedded in Pad Macro
11250100	DAT_RD_DLY2	32	**MSDC Data Delay Line Register 2** This register can configure the delay line embedded in Pad Macro
11250104	DAT_RD_DLY3	32	**MSDC Data Delay Line Register 3** This register can configure the delay line embedded in Pad Macro

11250110	**HW_DBG_SEL**	32	**MSDC H/W Debug Selection Register** This register can select the H/W debug output
11250114	**MAIN_VER**	32	**MSDC Main Version Register** This register shows the version code of MSDC IP
11250118	**ECO_VER**	32	**MSDC ECO Version Register** This register shows the ECO version code of MSDC IP

11250000 MSDC_CFG MSDC Configuration Register 02000099

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							SCLKSTOPSEL				CCKMD		CCKDIV[11:8]			
Type							RW				W1C		RW			
Reset							1				0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCKDIV[7:0]								CCKSB	BV18PSS	BV18SDT	CCKDRVE	PIO	RST	CCKPD	MSDC
Type	RW								RU	RU	RW	RW	RW	A0	RW	RW
Reset	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1

Bit(s)	Mnemonic	Name	Description
25	**SCLKSTOPSEL**	SCLK_STOP_SEL	**In DDR mode,stop SCLK when device is idle whether check SCLK phase** 1'b0: stop sclk no check SCLK phase 1'b1: stop sclk check SCLK phase,fix 1/4T sclk glitch in DDR mode
21:20	**CCKMD**	CARD_CK_MODE	**MS/SD Card clock mode** 2'b00: Use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b01: Use msdc_src_ck as msdc_ck, bit[15]~bit[8] is ignored. 2'b10: DDR mode, also use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b11: HS400 mode,also use clock divider output and use msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed.
19:8	**CCKDIV**	CARD_CK_DIV	**MS/SD Card clock divider** The register field controls clock frequency of serial clock on MS/SD bus. Please refer to Data Line Latching Timing Diagram and Response Latching Timing Diagram. For non-DDR mode, msdc_ck equals SD bus clock. (Ex: For SDR25 or HS, msdc_ck and SD bus clock will be 50MHz) For DDR mode, msdc_ck denotes the MSDC internal clock which will be dobule to SD bus clock. (Ex: For DDR50, msdc_ck should be set to 100MHz and bus clock will be 50MHz) 8'b00000000: msdc_ck =(1/2) * msdc_src_ck 8'b00000001: msdc_ck = (1/(4*1)) * msdc_src_ck 8'b00000010: msdc_ck = (1/(4*2)) * msdc_src_ck 8'b00000011: msdc_ck = (1/(4*3))* msdc_src_ck 8'b00010000: msdc_ck = (1/(4*16))* msdc_src_ck 8'b11111111: msdc_ck = (1/(4*255)) * msdc_src_ck
7	**CCKSB**	CARD_CK_STABLE	**MS/SD Card clock stable or not** After programming the CARD_CK_MODE or CARD_CK_DIV, this bit will immediately go to "0" and return to "1" if stable. User should poll this register to make sure the safety control of MSDC. 1'b0: Clock output is not stable 1'b1: Clock output is stable
6	**BV18PSS**	BV_1P8_PASS	**MSDC Bus voltage 1.8V detection status** S/W should check this bit after BUS_VOL_18V_START_DET turns

Bit(s)	Mnemonic	Name	Description
			to 0 from 1. 1'b0: The voltage detection has error. 1'b1: The voltage detection has no error.
5	**BV18SDT**	BV_1P8_START_DET	**MSDC Bus voltage 1.8V detection sequence start event** S/W writes this bit to 1 to trigger H/W outputs 1.8V clock for 1 ms and automatically detect CMD/DAT line sequence for voltage change is passed or not. H/Q will clear this bit to 0 after the detection has finished. The pass or fail status is stored in bit[6] BUS_VOL_18V_PASS.
4	**CCKDRVE**	CARD_CK_DRV_EN	**SD/MS Card Bus Clock drive enable bit** Set this bit to 1 to enable MSDC bus clock driver. The default bus state depends on MSDC_CFG][1] CARD_CK_PWDN bit. If MSDC_CFG[1] CARD_CK_PWDN= 1, the default clock state is free running. If MSDC_CFG[1] CARD_CK_PWDN = 0, the default clock state is gated to 0. Set this bit to 0 will put the bus state into "tri-state". Default is 1. 1'b0: Put the clock pad into tri-state 1'b1: Enable MSDC to drive clock pad, the state of CLK depends on MSDC_CFG[1] CARD_CK_PWDN
3	**PIO**	PIO_MODE	**MS/SD PIO mode** PIO mode selection. Default is in PIO mode. 1'b0: DMA mode 1'b1: PIO mode
2	**RST**	RST	**Software reset** Writing 1 to this register will cause internal synchronous reset of MS/SD controller, and it will not reset register settings and DMA controller. The reset sequence is done when this bit goes to 0. S/W should wait this bit back to 0 after writing 1. 1'b0: MS/SD controller is not in reset state 1'b1: MS/SD controller is in reset state
1	**CCKPD**	CARD_CK_PWDN	**MSDC bus clock power down mode** This bit controls the card clock power down mode. 1'b0: Clock is gated to 0 if no command or data is transmitted. 1'b1: Clock is free running even if no command or data is transmitted. (The clock may still be stopped when MSDC write data is not enough or no space for next read data)
0	**MSDC**	MSDC	**MS/SD mode selection** The register bit is used to configure the controller as the host of Memory Stick or as the host of SD/MMC Memory card. The default value is to configure the controller as the host of Memory Stick. 1'b0: Configure the controller as the host of Memory Stick 1'b1: Configure the controller as the host of SD/MMC Memory card

11250004 MSDC_IOCON MSDC IO Configuration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									RD7SPL	RD6SPL	RD5SPL	RD4SPL	RD3SPL	RD2SPL	RD1SPL	RD0SPL
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			WD3S	WD2S	WD1S	WD0S	WDSP	WDSP			RDSP	DDR5	DDLS	RDSP	RSPL	SDR1

			PL	PL	PL	PL	LSEL	L			LSEL	0CKD	EL	L		04CKS
Type			RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23	**RD7SPL**	R_D7_SMPL	**Read data 7 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
22	**RD6SPL**	R_D6_SMPL	**Read data 6 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
21	**RD5SPL**	R_D5_SMPL	**Read data 5 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
20	**RD4SPL**	R_D4_SMPL	**Read data 4 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
19	**RD3SPL**	R_D3_SMPL	**Read data 3 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
18	**RD2SPL**	R_D2_SMPL	**Read data 2 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
17	**RD1SPL**	R_D1_SMPL	**Read data 1 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
16	**RD0SPL**	R_D0_SMPL	**Read data 0 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
13	**WD3SPL**	W_D3_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
12	**WD2SPL**	W_D2_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
11	**WD1SPL**	W_D1_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
10	**WD0SPL**	W_D0_SMPL	**CRC Status and SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock

Bit(s)	Mnemonic	Name	Description
			falling edge
9	**WDSPLSEL**	W_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in write transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.W_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.W_D0_SMPL Data line 1: MSDC_IOCON.W_D1_SMPL Data line 2: MSDC_IOCON.W_D2_SMPL Data line 3: MSDC_IOCON.W_D3_SMPL
8	**WDSPL**	W_D_SMPL	**CRC Status and SDIO interrupt sample selection** 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
5	**RDSPLSEL**	R_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in read transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.R_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.R_D0_SMPL Data line 1: MSDC_IOCON.R_D1_SMPL Data line 2: MSDC_IOCON.R_D2_SMPL Data line 3: MSDC_IOCON.R_D3_SMPL Data line 4: MSDC_IOCON.R_D4_SMPL Data line 5: MSDC_IOCON.R_D5_SMPL Data line 6: MSDC_IOCON.R_D6_SMPL Data line 7: MSDC_IOCON.R_D7_SMPL
4	**DDR50CKD**	DDR50_DLY_SEL	**DDR50 output clock delay selection** 1'b0: Use default clock output 1'b1: Delay 1T msdc_src_ck for clock output
3	**DDLSEL**	D_DLYLINE_SEL	**Data line delay line fine tune selection** 1'b0: All data line share one delay selection value indicated by PAD_TUNE.PAD_DAT_RD_RXDLY 1'b1: Each data line has its own delay selection value indicated by Data line 0: DAT_RD_DLY0.DAT0_RD_DLY Data line 1: DAT_RD_DLY0.DAT1_RD_DLY Data line 2: DAT_RD_DLY0.DAT2_RD_DLY Data line 3: DAT_RD_DLY0.DAT3_RD_DLY Data line 4: DAT_RD_DLY1.DAT4_RD_DLY Data line 5: DAT_RD_DLY1.DAT5_RD_DLY Data line 6: DAT_RD_DLY1.DAT6_RD_DLY Data line 7: DAT_RD_DLY1.DAT7_RD_DLY
2	**RDSPL**	R_D_SMPL	**Read data sample selection** 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
1	**RSPL**	R_SMPL	**Command response sample selection** 1'b0: Sample response by external bus clock rising edge 1'b1: Sample response by external bus clock falling edge
0	**SDR104CKS**	SDR104_CLK_SEL	**SDR104 SCLK output clock control** This bit is only used when MSDC_CFG[17:16] CARD_CK_MODE is 2'b01. 1'b0: Bus clock output equals inverted msdc_src_ck 1'b1: Bus clock output equals msdc_src_ck

11250008 MSDC_PS MSDC Pin Status Register 01FF0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SDWP							CMD	DAT							
Type	RU							RU	RU							
Reset	0							1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CDDBCE														CDSTS	CDEN
Type	RW														RU	RW
Reset	0	0	0	0											1	0

Bit(s)	Mnemonic	Name	Description
31	**SDWP**	SD_WP	**Write Protection Switch status on SD Memory Card** The register bit shows the status of Write Protection Switch on SD Memory Card. There is no default reset value. The pin WP (Write Protection) is only useful while the controller is configured for SD Memory Card 1'b0: Write Protection Switch ON. It means that memory card is desired to be write-protected 1'b1: Write Protection Switch OFF. It means that memory card is writable
24	**CMD**	CMD	**Command line status** This bit reflects the command line value of MSDC bus.
23:16	**DAT**	DAT	**Data line status** This bit reflects the data line value of MSDC bus. (8-bits)
15:12	**CDDBCE**	CDDEBOUNCE	**Card detection de-bounce timer** The register field specifies the time interval for card detection de-bounce. Its default value is 0. It means that de-bounce interval is one 32KHz cycle. The interval will extend one cycle time of 32KHz by increasing the counter by 1
1	**CDSTS**	CDSTS	**Card detection status** 1'b1: Card detection pin status is logic high 1'b1: Card detection pin status is logic high
0	**CDEN**	CDEN	**Card detect enable** The register bit is used to control the card detection circuit 1'b0: Card detection is disable 1'b1: Card detection is enable

1125000C MSDC_INT MSDC Interrupt Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										AC53FAIL	AC53DONE	GEAROUTBOUND	DMAPROTECT	GPDCSERR	BDCSERR	AC19DONE
Type										W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SDDCRCERR	SDDTO	DMAXFDNE	SDXFCPL	SDCSTS	SDRCRCER	SDCTO	SDCRDY	SDIOIRQ	DMAQEPTY	SDACDRCRCER	SDACDCTO	SDACDCRDY		MSDCCDSC	MMCIRQ
Type	W1C	W1C	W1C	W1C	RU	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C		W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**AC53FAIL**	AUTOCMD53_FAIL	**SD Auto command 53 status register**

Bit(s)	Mnemonic	Name	Description
			When auto-command 53 tuning sequence is failure , this register will be set to 1 by H/W.
21	**AC53DONE**	AUTOCMD53_DONE	**SD Auto command 53 status register** When auto-command 53 tuning sequence is finished , this register will be set to 1 by H/W.
20	**GEAROUTBOUND**	GEAR_OUT_BOUND	**the gear setting of delayline is out of boundary during SDIO Autocmd 53 On-Line tuning process**
19	**DMAPROTECT**	DMA_PROTECT	**there is write operation to DMA start address, length, start bit or last buf bit**
18	**GPDCSERR**	GPD_CS_ERR	**GPD checksum error detected**
17	**BDCSERR**	BD_CS_ERR	**BD checksum error detected**
16	**AC19DONE**	AUTOCMD19_DONE	**SD Auto command 19 status register** When auto-command 19 is enabled, H/W will clear this register to 0. As the tuning sequence is finished (32 times), this register will be set to 1 by H/W. S/W should check the AUTOCMD_STS0 and SUTOCMD_STS1 only when this bit is ON.
15	**SDDCRCERR**	SD_DATA_CRCERR	**SD Data CRC error interrupt** Indicates that MS/SD controller detects a CRC error after reading a block of data from the DAT line or SD/MMC signals a CRC error after writing a block of data to the DAT line. 1'b0: Otherwise 1'b1: MS/SD controller detected a CRC error after reading a block of data from the DAT line or SD/MMC signaled a CRC error after writing a block of data to the DAT line
14	**SDDTO**	SD_DATTO	**SD Data timeout interrupt** Indicates that SD/MMC controller detects a timeout condition while waiting for data token on the DAT line. This bit is for both data read and data write. For SD data read, timeout will occur when the read data is not presented. For SD data write, timeout will occur when the write data CRC status is not presented if PATCH_BIT[30] DETECT_WR_CRC_TIMEOUT = 1 1'b0: Otherwise 1'b1: SD/MMC controller detects a timeout condition while waiting for data token on the DAT line
13	**DMAXFDNE**	DMA_XFER_DONE	**DMA transfer done interrupt** The register bit indicates the status of data block transfer. 1'b0: Otherwise 1'b1: A data block was successfully transferred
12	**SDXFCPL**	SD_XFER_COMPLETE	**SD Data transfer complete interrupt** This bit indicates the transaction which contains data has completed. While performing tuning procedure (Execute Tuning is set to 1), SD_XFER_COMPLETE is not set to 1.
11	**SDCSTS**	SD_CSTS	**SD CSTA update interrupt** The register bit indicates any bit in the register SDC_CSTA is active, the register bit will be set to 1. S/W should clear the SDC_CSTA and this bit will be de-asserted automatically. 1'b0: No SD Memory Card interrupt 1'b1: SD Memory Card interrupt exists
10	**SDRCRCER**	SD_RESP_CRCERR	**SD Command CRC error interrupt** Indicates that SD/MMC controller detected a CRC error after reading a response from the CMD line. 1'b0: Otherwise

Bit(s)	Mnemonic	Name	Description
			1'b1: SD/MMC controller detected a CRC error after reading a response from the CMD line
9	**SDCTO**	SD_CMDTO	**SD Command timeout interrupt** Indicates that SD/MMC controller detected a timeout condition while waiting for a response on the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a timeout condition while waiting for a response on the CMD line
8	**SDCRDY**	SD_CMDRDY	**SD Command ready interrupt** For the command without response, the register bit will be 1 once the command completes on SD/MMC bus. For command with response without busy, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error. For command with response with busy in DAT0, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error and the DAT0 transited from busy to idle. 1'b0: Otherwise 1'b1: Command finish successfully without a CRC error
7	**SDIOIRQ**	SD_SDIOIRQ	**SD SDIO interrupt** This bit indicates the interrupt is sensed in the SDIO bus. 1'b0: No interrupt on SDIO bus 1'b1: Interrupt on SDIO bus
6	**DMAQEPTY**	DMA_Q_EMPTY	**DMA queue empty interrupt** This bit is used to indicate the current DMA queue is empty. Only for Descriptor mode and Enhance mode.
5	**SDACDRCRCER**	SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt** This bit is set when detecting a CRC error in the Auto command response.
4	**SDACDCTO**	SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt** This bit is set if no response is returned within a specified cycles(64T in spec) from the end bit of Auto command.
3	**SDACDCRDY**	SD_AUTOCMD_CMDRDY	**SD auto command ready interrupt** This bit is set if auto command is executed without CRC error or time out.
1	**MSDCCDSC**	MSDC_CDSC	**MSDC Card detection status change interrupt** The register bit indicates if any interrupt for memory card insertion/removal exists. Whenever memory card is inserted or removed and card detection circuit is enabled, i.e., the register bit CDEN in the register MSDC_PS is set to 1, the register bit will be set to 1. It will be reset when the register is read. 1'b0: Otherwise 1'b1: Card is inserted or removed
0	**MMCIRQ**	MMC_IRQ	**MMC card interrupt** 1'b0: Otherwise 1'b1: indicates that MMC card interrupt event occurs

11250010 MSDC_INTEN MSDC Interrupt Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										ENAC53FAIL	ENAC53DONE	ENGEAROUTBOU	ENDMAPROTECT	ENGPDCSERR	ENBDCSERR	ENAC19DONE

												ND				
Type										RW	RW	RW	RW	RW	RW	RW
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENSDDCRCERR	ENSDDTO	ENDMAXFDNE	ENSDXFCPL	ENSDCSTA	ENSDRCRCER	ENSDCTO	ENSDCRDY	ENSDIOIRQ	ENDMAQEPTY	ENSDACDRCRCER	ENSDACDCTO	ENSDACDCRDY		ENMSDCCDSC	ENMMCIRQ
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
22	**ENAC53FAIL**	EN_AUTOCMD53_FAIL	**Auto-command 53 failure interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
21	**ENAC53DONE**	EN_AUTOCMD53_DONE	**Auto-command 53 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
20	**ENGEAROUTBOUND**	EN_GEAR_OUT_BOUND	**Gear out boundary interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
19	**ENDMAPROTECT**	EN_DMA_PROTECT	**DMA protection interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
18	**ENGPDCSERR**	EN_GPD_CS_ERR	**GPD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
17	**ENBDCSERR**	EN_BD_CS_ERR	**BD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
16	**ENAC19DONE**	EN_AUTOCMD19_DONE	**Auto-command 19 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
15	**ENSDDCRCERR**	EN_SD_DATA_CRCERR	**SD Data CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
14	**ENSDDTO**	EN_SD_DATTO	**SD Data timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
13	**ENDMAXFDNE**	EN_SD_DMA_XFER_DONE	**DMA transfer done interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
12	**ENSDXFCPL**	EN_SD_XFER_COMPLETE	**SD Data transfer complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
11	**ENSDCSTA**	EN_SD_CSTA	**SD CSTA update interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
10	**ENSDRCRCERR**	EN_SD_RESP_CRCERR	**SD Command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
9	**ENSDCTO**	EN_SD_CMDTO	**SD Command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

Bit(s)	Mnemonic	Name	Description
8	**ENSDCRDY**	EN_SD_CMDRDY	**SD Command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
7	**ENSDIOIRQ**	EN_SD_SDIOIRQ	**SD SDIO interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
6	**ENDMAQEPTY**	EN_DMA_Q_EMPTY	**DMA queue empty interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
5	**ENSDACDRCRCER**	EN_SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
4	**ENSDACDCTO**	EN_SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
3	**ENSDACDCRDY**	EN_AUTOCMD_CMDRDY	**SD auto command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
1	**ENMSDCCDSC**	EN_MSDC_CDSC	**MSDC Card detection status change interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
0	**ENMMCIRQ**	EN_MMC_IRQ	**MMC card interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt

11250014 MSDC_FIFOCS MSDC FIFO Control and Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFOCLR								TXFIFOCNT							
Type	A0								RU							
Reset	0								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXFIFOCNT							
Type									RU							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**FIFOCLR**	FIFOCLR	**Embedded FIFO clear** Write this bit to 1 makes FIFO cleared. It will goes to 0 when FIFO is cleared. S/W needs to check this bit to make sure clearing FIFO sequence is done. This bit can be used when the data read/write sequence has error and need to clean the H/W FIFO.
23:16	**TXFIFOCNT**	TXFIFOCNT	**TX FIFO count for MSDC write** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved
7:0	**RXFIFOCNT**	RXFIFOCNT	**RX FIFO count for MSDC read** 8'd0: No data in FIFO

Bit(s)	Mnemonic	Name	Description
			8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved

11250018 MSDC_TXDATA MSDC TX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIOTXDATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIOTXDATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIOTXDATA**	PIO_TXDATA	**PIO mode TXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

1125001C MSDC_RXDATA MSDC RX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIORXDATA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIORXDATA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIORXDATA**	PIO_RXDATA	**PIO mode RXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

11250030 SDC_CFG SD Configuration Register 00100000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DTOC										INTBGP	SDIODE	SDIO		BUSWD	
Type	RW										RW	RW	RW		RW	
Reset	0	0	0	0	0	0	0	0			0	1	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ENWKUPINS	ENWKUPSDIOINT
Type															RW	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31:24	**DTOC**	DTOC	**Data Timeout Counter** The period from the end of the initial host read command or the last read data block in a multiple block read operation to the start bit of the next read data block requires at least two serial clock cycles. The counter is used to extend the period (Read Data Access Time) in unit of 1048576 serial clocks. 8'b00000000: Extend 1048576 more serial clock cycle 8'b00000001: Extend 1048576x2 more serial clock cycle 8'b00000010: Extend 1048576x3 more serial clock cycle 8'b11111111: Extend 1048576x 256 more serial clock cycle
21	**INTBGP**	INT_AT_BLOCK_GAP	**Interrupt at block Gap** This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. Setting to 0 disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the Host Driver detects an SD card insertion, it shall set this bit according to the CCCR of the SDIO card. 1'b0: Disables interrupt detection at the block gap 1'b1: Enables interrupt detection at the block gap
20	**SDIOIDE**	SDIO_INT_DET_EN	**SDIO interrupt detection enable** This bit is to inform the SD controller to sense the SDIO interrupt 1'b0: SDIO interrupt detection is disabled 1'b1: SDIO interrupt detection is enabled if the SDIO bit is also on
19	**SDIO**	SDIO	**SDIO mode enable bit** This bit is to enable the support to sense the SDIO interrupt and disable the R4 response CRC check for SDIO card 1'b0: SDIO mode is disabled 1'b1: SDIO mode is enabled
17:16	**BUSWD**	BUSWIDTH	**Bus width configuration** This field is used to define the SD/MMC bus width 2'b00: 1 bit mode 2'b01: 4 bit mode 2'b10: 8 bit mode 2'b11: reserved
1	**ENWKUPINS**	WAKEUP_INS_EN	**Card status change wakeup event enable bit** 1'b0: Disable wakeup event for card status change 1'b1: Enable wakeup event for card status change
0	**ENWKUPSDIOINT**	WAKEUP_SDIOINT_EN	**SDIO card interrupt wakeup event enable bit** 1'b0: Disable wakeup event for SDIO card interrupt 1'b1: Enable wakeup event for SDIO card interrupt

11250034 SDC_CMD SD Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD53	VOLSWTH	ACMD		LEN											
Type	RW	RW	RW		RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOIRQ	STOP	RW	DTYPE			RSPTYP			BREAK	CMD					
Type	RW	RW	RW	RW			RW			RW	RW					
Reset	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**ACMD53**	AUTO_CMD53	**Auto command 53 enable** This field should use exclusively with AUTO_CMD field. means that AUTO_CMD != 2'b00, this field should assign to 1'b0, if this filed is assigned to 1'b1, AUTO_CMD should set to 2'b00 1'b0: Disable Auto Command 53 1'b1: Enable Auto Command 53
30	**VOLSWTH**	VOL_SWTH	**Voltage switch command** 1'b0: Disable voltage switch detection 1'b1: Enable voltage switch detection
29:28	**ACMD**	AUTO_CMD	**Auto command enable** This field determines use of auto command functions. This function can be used in all modes including PIO/Basic DMA/Descriptor DMA/Enhanced Mode. There are two methods to stop Multiple-block read and write operation. (1) Auto CMD12 Enable Multiple-block read and write commands for memory require CMD12 to stop the operation. When ACMD-12 is used, MSDC issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the MSDC_INT register. The Host Driver shall not set this bit if the command does not require CMD12. In particular, secure commands defined in the Part 3 File Security specification do not require CMD12. (2) Auto CMD23 Enable When ACMD-23 is used, MSDC issues a CMD23 automatically before issuing a command specified in the CMD field. The Host Controller Version 3.00 and later shall support this function. By writing the Command register, MSDC issues a CMD23 first and then issues a command specified by the CMD field in SDC_CMD register. If response errors of CMD23 are detected, the second command is not issued. A CMD23 error is indicated in the MSDC_INT register. 32-bit block count value for CMD23 is set to SDC_BLOCK_NUM register. 2'b00: Disable Auto Command 2'b01: Enable Auto CMD12 2'b10: Enable Auto CMD23 2'b11: Enable Re-tuning CMD19
27:16	**LEN**	LEN	**Length** The register field is used to define the length of one block in unit of byte in a data transaction of block mode or the data length in unit of byte in data transaction of byte mode. The maximal value of block length is 2048 bytes. 12'b000000000000: Reserved 12'b000000000001: Block length is 1 byte 12'b000000000010: Block length is 2 byte 12'b011111111111: Block length is 2047 byte 12'b100000000000: Block length is 2048 byte
15	**GOIRQ**	GO_IRQ	**GO_IRQ command** The register bit indicates if the command is GO_IRQ_STATE (CMD40) and used only for MMC protocol. If the command is GO_IRQ_STATE, the period between command token and response token will not be limited. 1'b0: The command is not GO_IRQ_STATE 1'b1: The command is GO_IRQ_STATE
14	**STOP**	STOP	**Stop command** The register bit indicates if the command is a stop transmission command. It should be set to 1 when CMD12 (SD/MMC) or CMD52

Bit(s)	Mnemonic	Name	Description
			with I/O abort (SDIO) is to be issued. 1'b0: The command is not a stop transmission command 1'b1: The command is a stop transmission command
13	**RW**	RW	**Command read write selection** The register bit defines the command is a read command or write command. The register bit is valid only when the command will cause a transaction with data token. 1'b0: The command is a read command 1'b1: The command is a write command
12:11	**DTYPE**	DTYPE	**Data block selection** The register field defines data token type for the command. 2'b00: No data token for the command 2'b01: Single block transaction (only available in block mode) 2'b10: Multiple block transaction. (only available in block mode) 2'b11: Stream operation. It only shall be used in MMC protocol. (only available in block mode)
9:7	**RSPTYP**	RSPTYP	**Command response type** 3'b000: This command has no response. 3'b001: The command has R1/R5/R6/R7 response. The response token is 48-bit with CRC check (For SD/MMC/SDIO) (Not include the SDIO abort command) 3'b010: The command has R2 response. The response token is 136-bit (For SD/MMC) 3'b011: The command has R3 response. The response token is 48-bit response, no CRC check (For SD/MMC) 3'b100: The command has R4 response. The response token is 48-bit without CRC check (For SDIO) The response token is 48-bit with CRC check (For MMC) 3'b111: The command has R1b response. The response token is 48-bit (For SD/MMC/SDIO)
6	**BREAK**	BREAK	**Abort a pending MMC GO_IRQ command** It is only valid for a pending GO_IRQ_MODE command waiting for MMC interrupt response. 1'b0: Not a beak command 1'b1: Break a pending MMC GO_IRQ_MODE command in the controller.
5:0	**CMD**	CMD	**SD Memory Card command**

11250038 SDC_ARG SD Argument Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARG[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ARG[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ARG**	ARG	**Memory card controller argument register**

1125003C SDC_STS SD Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMCSWRCPL															CMD_WR_BUSY
Type	RU															W1C
Reset	0															0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															CMDBSY	SDCBSY
Type															RU	RU
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31	**MMCSWRCPL**	MMC_STREAM_WR_COMPL	**MMC Stream mode write data is all flushed to MMC card** S/W can use this bit to confirm last write data are flushed to MMC then issue STOP command. This bit is only valid when the command SDC_CMD.DTYPE=2'b11. 1'b0: Last Data are partially inside MSDC 1'b1: Last data are flushed to MMC card
16		CMD_WR_BUSY	
1	**CMDBSY**	CMDBUSY	**SD Command line busy status** S/W should always read this bit to make sure the command line is not busy before sending the next command. If the command is R1B or data read/write command, S/W should check SDCBUSY bit too. Note: When Auto command 12 is enabled, this bit will be asserted immediately after SDC_CMD is written and de-asserted after auto-command 12 finishes. 1'b0: No transmission is going on CMD line on SD bus 1'b1: There exists transmission going on CMD line on SD bus
0	**SDCBSY**	SDCBUSY	**SD controller busy status** 1'b0: SD controller is idle 1'b1: SD controller is busy

11250040 SDC_RESP0 SD Response Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP0**	RESP0	**Memory card controller response register 0**

11250044 SDC_RESP1 SD Response Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP1[31:16]															
Type	RU															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP1[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP1**	RESP1	**Memory card controller response register 1**

11250048 SDC_RESP2 SD Response Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP2[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP2[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP2**	RESP2	**Memory card controller response register 2**

1125004C SDC_RESP3 SD Response Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP3[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP3[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP3**	RESP3	**Memory card controller response register 3**

11250050 SDC_BLK_NUM SD Block Number Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BLKNUM[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BLKNUM[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0	**BLKNUM**	BLOCK_NUMBER	**Memory card controller Block number** This field indicates the block number of data transaction.

Bit(s)	Mnemonic	Name	Description
			32'd0: Reserved 32'd1: 1 data block 32'd2: 2 data block 32'd3: 3 data block 32'hffffffff: 4GB-1 data block

11250054 SDC_VOL_CHG SD Voltage Change Wait Time Register 00000145

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VCHGCNT															
Type	RW															
Reset	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15:0	**VCHGCNT**	VOL_CHG_WAIT_CNT	**This register define SD voltage change check wait time,wait time is clock frequency multiply VOL_WAIT_TIME**

11250058 SDC_CSTS SD Card Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS**	CSTS	**CSTS** The card status field in the response R1 or R1b field. Each bit can be write 1 clear individually.

1125005C SDC_CSTS_EN SD Card Status Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS_EN[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS_EN[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS_EN**	CSTS_EN	**CSTS_EN** This register is used to control which bit of the CSTA will generate

Bit(s)	Mnemonic	Name	Description
			the MSDC_INT.SDCSTA

11250060 **SDC_DATCRC_STS** **SD Card Data CRC Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCSSN								DCSSP							
Type	RU								RU							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8	**DCSSN**	DAT_CRCSTS_NEG	**MSDC DDR mode negative edge Read DATA CRC status** This register reflects the CRC status of data line[7:0] in DDR mode. The positive edge CRC status is shown in DAT_CRC_STS[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error
7:0	**DCSSP**	DAT_CRCSTS_POS	**MSDC read DATA CRC status** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error

11250080 **SD_ACMD_RESP** **SD ACMD Response Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMDRESP[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMDRESP[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMDRESP**	AUTOCMD_RESP	**SD Auto command response register** This register stores the response[39:8] of ACMD12/ACMD23/ACMD19.

11250084 **SD_ACMD19_TRG** **SD ACMD19 Target Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ACMDFTSEL			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**ACMDFTSEL**	FINE_TUNE_SEL	**SD Auto command 19 test target selection** After auto-command 19 is triggered, MSDC will only change the phase of the selected one and keep the value of other tuning registers. S/W can get the result from AUTOCMD_STS. There are total 32 phases for each delay line. 4'd0: Select PAD_CLK_TXDLY[5:0] as the target to run auto-command19. 4'd1: Select PAD_CMD_RXDLY[5:0] as the target to run auto-command19. 4'd2: Select PAD_DAT_RD_RXDLY[5:0] as the target to run auto-command19. 4'd3: Select PAD_DAT_WR_RXDLY[5:0] as the target to run auto-command19. 4'd4: Select DAT0_RD_DLY[5:0] as the target to run auto-command19. 4'd5: Select DAT1_RD_DLY[5:0] as the target to run auto-command19. 4'd6: Select DAT2_RD_DLY[5:0] as the target to run auto-command19. 4'd7: Select DAT3_RD_DLY[5:0] as the target to run auto-command19. 4'd8: Select DAT4_RD_DLY[5:0] as the target to run auto-command19. 4'd9: Select DAT5_RD_DLY[5:0] as the target to run auto-command19. 4'd10: Select DAT6_RD_DLY[5:0] as the target to run auto-command19. 4'd11: Select DAT7_RD_DLY[5:0] as the target to run auto-command19. 4'd12: Select CMD_RESP_RXDLY[5:0] as the target to run auto-command19. Others: Reserved

11250088 **SD_ACMD19_STS** **SD ACMD19 Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD19STS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMD19STS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMD19STS**	AUTOCMD19_STS	**SD Auto command 19 test result register** When auto-command 19 is enabled, H/W will automatically try 32 times of command-19 and store the result into this register. This register contains 1st to 32th results in bit[0:31]

1125008C DMA_SA_HIGH 4BIT DMA Current Address Resgiter of high 4bit 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													DMASAHIGH4BIT			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**DMASAHIGH4BIT**	DMA_SURR_ADDR_HIGH4BIT	**it is usend to set high 4bit address of start address because 64G dram need 36bit address**

11250090 DMA_SA DMA Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMASA[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMASA[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMASA**	DMA_STR_ADDR	**The start address of the DMA address** This register is used to set the start address of the DMA. In DMA basic mode, this field indicates the source or destination address of the data transfer which depends on the command. In descriptor base DMA, this is the descriptor chain start address.

11250094 DMA_CA DMA Current Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMACA**	DMA_CURR_ADDR	**The current address of the DMA address** This register is used to read the current address of the DMA descriptor chain.

11250098 DMA_CTRL DMA Control Register 00006008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		**BSTSZ**			**SPLIT1K**	**LASTBF**	**DMAALIGN**	**DMAMOD**					**READYM**	**DMARSM**	**DMASTOP**	**DMASTART**
Type		RW			RW	RW	RW	RW					RO	WO	A0	WO
Reset		1	1	0	0	0	0	0					1	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**BSTSZ**	BURST_SIZE	**DMA burst size** This field is used to specify the maximum transfer bytes allowed at the device per DMA burst. This field can not be modified when the DMA status is 1. 3'd3: 8 Bytes 3'd4: 16 Bytes 3'd5: 32 Bytes 3'd6: 64 Bytes Other: Reserved
11	**SPLIT1K**	DMA_SPLIT_1K	**This field is used to specify whether split burst when corss 1K boundry address** 1'b0: 1K boundary not split 1'b1: 1K boundary split
10	**LASTBF**	LAST_BUF	**Last buffer of the basic DMA mode** This field indicates the last buffer in the basic DMA mode
9	**DMAALIGN**	DMA_ALIGN	**This field is used to specify whether address aligment burst size** 1'b0: do not DAM burst size alignment 1'b1: DAM burst size alignment
8	**DMAMOD**	DMA_MODE	**DMA operation mode** This field indicates operation mode of DMA 1'b0: Basic DMA mode 1'b1: Descriptor base DMA mode
3	**READYM**	AHB_READYM	**only for debug when dma hang,sw can check if ahb bus is ok when gdma is hang** 1: bus is normal 0: bus not normal
2	**DMARSM**	DMA_RESUME	**DMA resume control register** This bit is used to resume the DMA transaction. Read always return 0
1	**DMASTOP**	DMA_STOP	**DMA Stop control register** This bit is used to stop the DMA transaction. When SW issue STOP command, SW must wait this bit de-assert or DMA inactive to guarantee stop done.
0	**DMASTART**	DMA_START	**DMA start control register** This bit is used to start the DMA transaction. Read always return 0

1125009C DMA_CFG DMA Configuration Register 00000004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name															**OUTBOUND**	**DMACHKSU**

															STOPDMA	M12B
Type															RW	RW
Reset															0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MSDCACTIVEEN				AHBHPROT2EN							LOCKDISABLE	DSCPCSEN	DMASTS
Type			RW				RW							RW	RW	RU
Reset			0	0			0	0						1	0	0

Bit(s)	Mnemonic	Name	Description
17	**OUTBOUNDSTOPDMA**	OUT_BOUND_STOP_DMA	**This register will determine whether stop Enhance DMA if gear setting is out-of-boundary during send training data** 1'b0: Enhance DMA will continue even if gear setting is out-of-boundary during send training data 1'b1: Enhance DMA will stop if gear setting is out-of boundary during send training data
16	**DMACHKSUM12B**	DMA_CHK_SUM_12B	**This register indicates GPD/BD checksum cover 16byte or 12byte** 1'b0: GPD/BD checksum cover 16byte 1'b1: GPD/BD checksum only cover 12byte
13:12	**MSDCACTIVEEN**	MSDC_ACTIVE_EN	**This register will indicate how to control msdc_active** 2'b00: dynamic control msdc_active 2'b01: msdc_active = 0 2'b10: msdc_active = 1 2'b11: Reserved
9:8	**AHBHPROT2EN**	AHB_HPROT_2_EN	**This register will determine how to control hprot_2 pin of AHB bus** AHB_HPROT_2_EN = 2'b00, and Basic DMA Mode All the write transfers of a burst will access by bufferable mode except the last burst of DMA AHB_HPROT2_2_EN=2'b00, and Descriptor DMA Mode all the write transfers of a burst will access by bufferable mode except HW own update transfer 2'b00: dynamic control hprot_2 2'b01: hprot_2 = 0 2'b10: hprot_2 = 1
2	**LOCKDISABLE**	LOCK_DISABLE	**should disable lock in order to improve emi efficient** 1'b0: enable ahb lock 1'b1: disable ahb lock
1	**DSCPCSEN**	DMA_DSCP_CS_EN	**DMA descriptor checksum enable** This bit is used to enable or disable the descriptor checksum validation function for the descriptor. This field can not be modified when the DMA status is 1.
0	**DMASTS**	DMA_STATUS	**DMA status** This bit is used to indicate the status of the DMA. 1'b0: DMA engine is inactive 1'b1: DMA engine is active

112500A0 SW_DBG_SEL MSDC S/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGSEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**SWDBGSEL**	DBG_SEL	**MSDC debug selection** This contain is reserved!

112500A4 SW_DBG_OUT MSDC S/W Debug Output Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SWDBGO[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGO[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SWDBGO**	DBG_OUT	**MSDC debug output** 32 bit output selected by SW_DBG_SEL register

112500A8 DMA_LENGTH DMA Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	XFSZ[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	XFSZ[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**XFSZ**	XFER_SIZE	**DMA total transfer size** This field is used to specify the number of DMA transfer byte required for the movement of source data through DMA. This field is only valid in basic DMA mode.

112500B0 PATCH_BIT0 MSDC Patch Bit Register 0 403C0006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PTCH 31	PTCH 30	PTCH 29	PTCH 28	PTCH 27	PTCH 26	PTCH22				PTCH18				PTCH 17	PTCH 16
Type	RW	RW	RW	RW	RW	RW	RW				RW				RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PTCH 15	CKGDLYS					INTCKS			DESC UP	PTCH 5	PTCH 4	PTCH 3	PTCH 02	PTCH 01	
Type	RW	RW					RW			RW	RW	RW	RW	RW	RW	

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	

Bit(s)	Mnemonic	Name	Description
31	**PTCH31**	EN_MMC_DRV_RESP	**Enable MSDC always drives bus when output wakeup response (BREAK)** 1'b0: Disable 1'b1: Enable
30	**PTCH30**	DETECT_WR_CRC_TIMEOUT	**MSDC write data CRC phase timeout detection** 1'b0: Not detect CRC phase timeout 1'b1: detect CRC phase timeout
29	**PTCH29**	SPC_ALWAYS_PUSH	**SPC Buffer push mechanism** 1'b0: Push the buffer only when read transfer is on-going 1'b1: Always push the buffer
28	**PTCH28**	SDIO_INT_DLY_SEL	**SDIO interrupt latch time selection** 1'b0: Latch the data line value in internal SDIO interrupt period 1'b1: Latch the data line value in 1 clock delay of internal SDIO interrupt period
27	**PTCH27**	SDC_CMD_CMDFAIL_SEL	**SDIO interrupt period recovery selection** 1'b0: SDIO interrupt period will re-start after a CMD12 or CMD52 command is issued 1'b1: SDIO interrupt period whenever DAT line is not busy
26	**PTCH26**	SDC_CMD_IDRT_SEL	**SD identification response time selection** The register bit indicates if the command has a response with NID (that is, 5 serial clock cycles as defined in SD Memory Card Specification Part 1 Physical Layer Specification version 1.0) response time. The register bit is valid only when the command has a response token. Thus the register bit must be set to 1 for CMD2 (ALL_SEND_CID) and ACMD41 (SD_APP_OP_CMD). 1'b0: Otherwise. 1'b1: The command has a response with NID response time.
25:22	**PTCH22**	SDC_CFG_WDOD	**SD Write Data Output Delay** The period from finish of the response for the initial host write command or the last write data block in a multiple block write operation to the start bit of the next write data block requires at least two serial clock cycles. The register field is used to extend the period (Write Data Output Delay) in unit of one serial clock. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
21:18	**PTCH18**	SDC_CFG_BSYDLY	**SD R1B busy detection mode** The register field is only valid for the commands with R1b response. If the command has a response of R1b type, MS/SD controller must monitor the data line 0 for card busy status from the bit time that is two serial clock cycles after the command end bit to check if operations in SD/MMC Memory Card have finished. The register field is used to expand the time between the command end bit and end of detection period to detect card busy status. If time is up and there is no card busy status on data line 0, then the controller will abandon the detection. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
17	**PTCH17**	SDIO_CFG_INTC_SEL	**SDIO Interrupt model selection** 1'b0: Only when data line [1] = 0 and then trigger SDIO interurpt event

Bit(s)	Mnemonic	Name	Description
			1'b1: Only when data line [3:0] = 4'b1101 and then trigger SDIO interurpt event
16	**PTCH16**	MSDC_BLKNUM_SEL	**Configuration support ACMD23 reliable/force prog etc. feature** 1'b0: Support ACMD23 reliable/force prog etc. feature 1'b1: Don't support ACMD23 reliable/force prog etc. feature
15	**PTCH15**	MSDC_FIFO_RD_DIS	**MSDC RXFIFO Read Disable** 1'b0: Disable FIFO read permission to RXFIFO in PIO mode 1'b1: Enable FIFO read permission to RXFIFO in PIO mode
14:10	**CKGDLYS**	CKGEN_MSDC_DLY_SEL	**CKBUF in CKGEN Delay Selection** Total 32 stages
9:7	**INTCKS**	INT_DAT_LATCH_CK_SEL	**Internal MSDC clock phase selection** Total 8 stages, each stage can delay 1 clock period of msdc_src_ck
6	**DESCUP**	DESC_UP_SEL	**sd transfer done int should be issue when GPD have been update** 1'b1: enable new function for generate sd transfer done int 1'b0: use old function for generate sd transfer done int
5	**PTCH5**	ACMD53_FAIL_ONE_SHOT	**determine interrupt method of AUTOCMD53_FAIL** 1'b0: AUTOCMD53_FAIL interrupt will assert whenever CMD/DAT crc error occur 1'b1: AUTOCMD53_FAIL interrupt will assert only when AUTOCMD53_DONE assert if there is CMD/DAT crc error
4	**PTCH4**	MASK_ACMD53_CRC_ERR_INTR	**mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence** 1'b0: enable CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence 1'b1: mask CMD/DAT crc error interrupt during execute AUTOCMD53 training sequence
3	**PTCH3**	RD_DAT_SEL	**This field is used to define whther used rising or falling buf data for SDR mode** 1'b0: Used rising buf data for SDR mode 1'b1: Used falling buf data for SDR mode
2	**PTCH02**	DIS_REFLECT_CMDWR_WHEN_BSY	**Enable SD command register write montior** 1'b0: Enable monitor function 1'b1: Disable monitor function
1	**PTCH01**	EN_SDC_ODD_8BIT_SUP	**Enable SD odd number support for 8-bit data bus** 1'b0: Disable 1'b1: Enable

112500B4 **PATCH_BIT1** **MSDC Patch Bit Register 1** **FFB00009**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSHBFCKEN	MRCTLCKEN	MWCTLCKEN	MSDCKEN	MACMDCKEN	MVOLDTCKEN	MPSCCKEN	MSPCCKEN	HGDMACKEN		DCMEN	DCMDIVSEL1				SINGLEBURST
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW				RW
Reset	1	1	1	1	1	1	1	1	1		1	1				0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESETGDMA		BIAS28R0	BIAS28R1	BIAS28R2				GETCRCMARGIN	GETBUSYMARGIN	CMDTA			WRTA		
Type	RW		RW	RW	RW				RW	RW	RW			RW		
Reset	0		0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
31	**MSHBFCKEN**	MSDC_CK_SHBFF_CKEN	**msdc_src_ck clock enable bit for SHBFF** 1'b0: Disable 1'b1: Enable
30	**MRCTLCKEN**	MSDC_CK_RCTL_CKEN	**msdc_src_ck clock enable bit for RCTL** 1'b0: Disable 1'b1: Enable
29	**MWCTLCKEN**	MSDC_CK_WCTL_CKEN	**msdc_src_ck clock enable bit for WCTL** 1'b0: Disable 1'b1: Enable
28	**MSDCKEN**	MSDC_CK_SD_CKEN	**msdc_src_ck clock enable bit for SD** 1'b0: Disable 1'b1: Enable
27	**MACMDCKEN**	MSDC_CK_ACMD_CKEN	**msdc_src_ck clock enable bit for ACMD** 1'b0: Disable 1'b1: Enable
26	**MVOLDTCKEN**	MSDC_CK_VOLDET_CKEN	**msdc_src_ck clock enable bit for VOLDET** 1'b0: Disable 1'b1: Enable
25	**MPSCCKEN**	MSDC_CK_PSC_CKEN	**msdc_src_ck clock enable bit for PSC** 1'b0: Disable 1'b1: Enable
24	**MSPCCKEN**	MSDC_CK_SPC_CKEN	**msdc_src_ck clock enable bit forSPC** 1'b0: Disable 1'b1: Enable
23	**HGDMACKEN**	AHB_CK_GDMA_CKEN	**hclk_ck clock enable bit for GDMA** 1'b0: Disable 1'b1: Enable
21	**DCMEN**	DCM_EN	**host enable dcm for low power,when there is no transfer the ahb_ck and msdc_src_ck will divder from spurce clk** 1'b1: disable 1'b0: enable
20	**DCMDIVSEL1**	DCM_DIV_SEL1	**for spec4.5 divider msdc_src_ck from source clk,for spec50 divider ahb_ck from source clk** 1'b1: 32 divder from source clk 1'b0: 16 divder from source clk
16	**SINGLEBURST**	ENABLE_SINGLE_BURST	**the ahb bus will not support incr1 burst type in future.And it will only affect AHB bus msdc design,not affect AXI bus design** 1'b0: hw will send incr1 burst type 1'b1: hw will send single burst typr instead of incr1 type
15	**RESETGDMA**	RESET_GDMA	**sw can sw reset gdma when design hang** 1 'b1: reset gdma 1'b0: not reset gdma
13	**BIAS28R0**	BIAS_EXTBIAS_28NM	**28NM BIAS Controler register 0**
12	**BIAS28R1**	BIAS_EN18IO_28NM	**28NM BIAS Controler register 1**
11:8	**BIAS28R2**	BIAS_TUNE_28NM	**28NM BIAS Controler register 2**
7	**GETCRCMARGIN**	GET_CRC_MARGIN	**it will add margin for get crc status when card resp crc not match spec 2cycle from endbit** 1'b0: 8 cycle reserved for get crc status from write data crc endbit 1'b1: 16 cycle reserved for get crc status from write data crc endbit
6	**GETBUSYM**	GET_BUSY_MARG	**it will add margin for get busy state of data0**

Bit(s)	Mnemonic	Name	Description
	ARGIN	IN	1'b0: 1 cycle reserved for get busy state from src status endbit 1'b1: 3cycle reserved for get busy state from src status endbit
5:3	**CMDTA**	CMD_RSP_TA_CNTR	**CMD response turn around period** The turn around cycle = CMD_RSP_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode, this register should be set to 0
2:0	**WRTA**	WRDAT_CRCS_TA_CNTR	**Write data and CRC status turn around period** The turn around cycle = WRDAT_CRCS_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode,this register should be set to 0

112500B8 PATCH_BIT2 MSDC Patch Bit Register 2 14801803

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCSTSENSEL			CFGCRCSTS	CFGCRCSTSCNT		CFGCRCSTSEDGE	CFG_CRCSTS_SEL	POPENCNT					RESPSTSENSEL		
Type	RW			RW	RW		RW	RW	RW					RW		
Reset	0	0	0	1	0	1	0	0	1	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFGRESP	CFGRESPCNT			INTCRESPSEL		CFGRDAT	CFGRDATCNT					RESPWAITCNT		SUPPORT64G	ENHANCEGPD
Type	RW	RW			RW		RW	RW					RW		RW	RW
Reset	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:29	**CRCSTSENSEL**	CRCSTS_LATCH_EN_SEL	**This register used configuration latch CRC Status enable signal for async fifo in emmc45** 3'b000: latch CRC Status enable signal not delay 3'b001: latch CRC Status enable signal delay 1T msdc_ck 3'b010: latch CRC Status enable signal delay 2T msdc_ck 3'b011: latch CRC Status enable signal delay 3T msdc_ck 3'b111: latch CRC Status enable signal delay 7T msdc_ck
28	**CFGCRCSTS**	CFG_CRCSTS	**This register used configuration CRC Status path selection,this setting only used emmc4.5 feature** 1'b0: Latch CRC Status select delay-line path 1'b1: Latch CRC Status select async fifo path
27:26	**CFGCRCSTSCNT**	CFG_CRCSTS_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 2'b00: push 0 data in async fifo when start pop out data from async fifo 2'b01: push 1 data in async fifo when start pop out data from async fifo 2'b10: push 2 data in async fifo when start pop out data from async fifo 2'b11: push 3 data in async fifo when start pop out data from async fifo
25	**CFGCRCSTSEDGE**	CFG_CRCSTS_EDGE	**This register configuration used rising async fifo or falling async fifo** 1'b0: async fifo latch CRC Status used rising async fifo 1'b1: async fifo latch CRC Status used falling async fifo

Bit(s)	Mnemonic	Name	Description
24		CFG_CRCSTS_SEL	**This register configuration async fifo path selection** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
23:20	**POPENCNT**	POP_EN_CNT	**pop enable counter** This field is used to define how many write pointer and the read pointer margin began to pop data transfer
18:16	**RESPSTSENSEL**	RESP_LATCH_EN_SEL	**This register used configuration latch CMD Response enable signal for async fifo in emmc45** 3'b000: latch CMD Response enable signal not delay 3'b001: latch CMD Response enable signal delay 1T msdc_ck 3'b010: latch CMD Response enable signal delay 2T msdc_ck 3'b011: latch CMD Response enable signal delay 3T msdc_ck 3'b111: latch CMD Response enable signal delay 7T msdc_ck
15	**CFGRESP**	CFG_RESP	**This register used configuration CMD Response path selection,this setting only used emmc4.5 feature** 1'b0: Latch CMD Response select async fifo path 1'b1: Latch CMD Response select delay-line path
14:12	**CFGRESPCNT**	CFG_RESP_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 3'b000: push 0 data in async fifo when start pop out data from async fifo 3'b001: push 1 data in async fifo when start pop out data from async fifo 3'b111: push 7 data in async fifo when start pop out data from async fifo
11	**INTCRESPSEL**	INTC_RESP_SEL	**This register configuration BREAK command async fifo path** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
9	**CFGRDAT**	CFG_RDAT	**This register used configuration read data path** 1'b0: read data path by pass delay line 1'b1: read data path through delay line
8:4	**CFGRDATCNT**	CFG_RDAT_CNT	**This register used configuration read data path delay line**
3:2	**RESPWAITCNT**	RESP_WAIT_CNT	**This register used configuration cmd response timeout,timeout cycle is 65T+16*RESP_WAIT_CNT** 2'b00: cmd response timeout is 65T 2'b01: cmd response timeout is 65T+ 16*1T 2'b01: cmd response timeout is 65T+ 16*2T 2'b01: cmd response timeout is 65T+ 16*3T
1	**SUPPORT64G**	SUPPORT_64G	**This register used which proj support high 64G dram space access** 1'b1: support 64G dram access 1'b0: not support 64G dram access
0	**ENHANCEGPD**	ENHANCE_WAIT_GPD	**if sw clear int when gpd not update finish,design will hang.so you can set this bt to 1 to avoid this iuuse in enhance write mode** 1'b1: use new HW code for update gpd in enhance mode 1'b0: use old HW code

112500C0 DAT0_TUNE_C DAT0 Tune Result Register 00000000

RC

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT0CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT0CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT0CRCSTS**	DAT0_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT0 into this register. This register contains 1st to 32th results in bit[0:31]

112500C4 DAT1_TUNE_CRC DAT1 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT1CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT1CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT1CRCSTS**	DAT1_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT1 into this register. This register contains 1st to 32th results in bit[0:31]

112500C8 DAT2_TUNE_CRC DAT2 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT2CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT2CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DAT2CRCSTS**	DAT2_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT2 into this register.

Bit(s)	Mnemonic	Name	Description
			This register contains 1st to 32th results in bit[0:31]

112500CC DAT3_TUNE_CRC DAT3 Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT3CRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DAT3CRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	DAT3CRCSTS	DAT3_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of DAT3 into this register. This register contains 1st to 32th results in bit[0:31]

112500D0 CMD_TUNE_CRC CMD Tune Result Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CMDCRCSTS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CMDCRCSTS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	CMDCRCSTS	CMD_CRC_STS	**SDIO Auto command 53 test result register** When auto-command 53 is enabled, H/W will automatically try TUNE_WINDOW times of command-53 and store CRC result of CMD into this register. This register contains 1st to 32th results in bit[0:31]

112500D4 SDIO_TUNE_WIND SDIO Tune Window Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												TUNEWINDOW				
Type												RW				
Reset												0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4:0	**TUNEWINDOW**	TUNE_WINDOW	**SDIO Auto command 53 tuning window setting. the window range will be** 1. for CMD PAD low bound: max(0, PAD_CMD_RXDLY-TUNE_WINDOW) high bound: min(31, PAD_CMD_RXDLY+TUNE_WINDOW) 2. for DAT PAD low bound: max(0, DAT(0:3)_RD_DLY-TUNE_WINDOW) high bound:min(31: DAT(0:3)_RD_DLY+TUNE_WINDOW)

112500F0 PAD_TUNE0 MSDC Pad Tuning Register0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKTDLY					CMDRRDLY					CMDRRDLYSEL	CMDRDLY				
Type	RW					RW					RW	RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDLYSEL		DATRRDLYSEL	DATRRDLY					DELAYEN			DATWRDLY				
Type	RW		RW	RW					RW			RW				
Reset	0		0	0	0	0	0	0	0			0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:27	**CLKTDLY**	PAD_CLK_TXDLY	**CLK Pad TX Delay Control** This register is used to add delay to CLK phase. Total 32 stages
26:22	**CMDRRDLY**	PAD_CMD_RESP_RXDLY	**CMD Response Internal Delay Line Control** This register is used to fine-tune response phase latched by MSDC internal clock Total 32 stages
21	**CMDRRDLYSEL**	PAD_CMD_RD_RXDLY_SEL	**Decide CMD Response pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY**	PAD_CMD_RXDLY	**CMD Pad RX Delay Line1 Control** This register is used to fine-tune CMD pad macro respose latch timing Total 32 stages
15	**RXDLYSEL**	PAD_RXDLY_SEL	**Decide rx delay line tune data path or clock path** 1'b0: rx delay line tune data path 1'b1: rx delay line tune clock path
13	**DATRRDLYSEL**	PAD_DAT_RD_RXDLY_SEL	**Decide rx data pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY**	PAD_DAT_RD_RXDLY	**DAT Pad RX Delay Line1 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages
7	**DELAYEN**	DELAY_EN	**enable all delay cell toogle when power on** 1'b0: disable delay cell toogle default 1'b1: enable delay cell toogle default

Bit(s)	Mnemonic	Name	Description
4:0	**DATWRDLY**	PAD_DAT_WR_RXDLY	**Write Data Status Internal Delay Line Control** This register is used to fine-tune write status phase latched by MSDC internal clock Total 32 stages

112500F4 PAD_TUNE1 MSDC Pad Tuning Register1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											CMDRRDLY2SEL	CMDRDLY2				
Type											RW	RW				
Reset											0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DATRRDLY2SEL	DATRRDLY2												
Type			RW	RW												
Reset			0	0	0	0	0	0								

Bit(s)	Mnemonic	Name	Description
21	**CMDRRDLY2SEL**	PAD_CMD_RD_RXDLY2_SEL	**Decide CMD Response pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY2**	PAD_CMD_RXDLY2	**CMD Pad RX Delay Line2 Control** This register is used to fine-tune CMD pad macro respose latch timing in data path Total 32 stages
13	**DATRRDLY2SEL**	PAD_DAT_RD_RXDLY2_SEL	**Decide rx data pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY2**	PAD_DAT_RD_RXDLY2	**DAT Pad RX Delay Line2 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages

112500F8 DAT_RD_DLY0 MSDC Data Delay Line Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY								DAT1RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY								DAT3RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY**	DAT0_RD_DLY	**DAT0 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY**	DAT1_RD_DLY	**DAT1 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

Bit(s)	Mnemonic	Name	Description
12:8	**DAT2RDDLY**	DAT2_RD_DLY	**DAT2 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY**	DAT3_RD_DLY	**DAT3 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

112500FC DAT_RD_DLY1 MSDC Data Delay Line Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY								DAT5RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY								DAT7RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY**	DAT4_RD_DLY	**DAT4 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY**	DAT5_RD_DLY	**DAT5 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY**	DAT6_RD_DLY	**DAT6 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY**	DAT7_RD_DLY	**DAT7 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

11250100 DAT_RD_DLY2 MSDC Data Delay Line Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY2								DAT1RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY2								DAT3RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY2**	DAT0_RD_DLY2	**DAT0 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY2**	DAT1_RD_DLY2	**DAT1 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT2RDDLY2**	DAT2_RD_DLY2	**DAT2 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY2**	DAT3_RD_DLY2	**DAT3 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11250104 DAT_RD_DLY3 MSDC Data Delay Line Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY2								DAT5RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY2								DAT7RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY2**	DAT4_RD_DLY2	**DAT4 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY2**	DAT5_RD_DLY2	**DAT5 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY2**	DAT6_RD_DLY2	**DAT6 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY2**	DAT7_RD_DLY2	**DAT7 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

11250110 HW_DBG_SEL MSDC H/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBGWSEL	DBG3SEL						DBG2SEL							
Type		RW	RW						RW							
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DBG1SEL						DBG0SEL							
Type			RW						RW							
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30	**DBGWSEL**	HW_DBG_WRAP_SEL	**H/W debug output selection for wrapper** 0: Select original debug pins 1: Select wrapper debug pins
29:24	**DBG3SEL**	HW_DBG3_SEL	**H/W debug output selection**
23:16	**DBG2SEL**	HW_DBG2_SEL	**H/W debug output selection**
13:8	**DBG1SEL**	HW_DBG1_SEL	**H/W debug output selection**
7:0	**DBG0SEL**	HW_DBG0_SEL	**H/W debug output selection**

11250114 MAIN_VER MSDC Main Version Register 20140512

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINVER[31:16]															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAINVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:0	**MAINVER**	MAIN_VER	**Main Version**

11250118 ECO_VER MSDC ECO Version Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECOVER[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECOVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ECOVER**	ECO_VER	**ECO Version**

Module name: MSDC Base address: (+0h)

Address	Name	Width	Register Function
112c0000	**MSDC_CFG**	32	**MSDC Configuration Register** The register is for general configuration of the MS/SD controller.
112c0004	**MSDC_IOCON**	32	**MSDC IO Configuration Register** The register contains the receiver path data latch timing control and interface control bits.
112c0008	**MSDC_PS**	32	**MSDC Pin Status Register** The register is used to storing card detection and write protection pin status. Card detection status can be disabled.
112c000C	**MSDC_INT**	32	**MSDC Interrupt Register** The register contains the status of interrupts. Note that the register still shows the status of interrupt even though the interrupt is disabled.
112c0010	**MSDC_INTEN**	32	**MSDC Interrupt Enable Register** The register contains the related enable bit of interrupts.
112c0014	**MSDC_FIFOCS**	32	**MSDC FIFO Control and Status Register** The register contains the control and status of embedded 128B FIFO.
112c0018	**MSDC_TXDATA**	32	**MSDC TX Data Port Register** The register is for PIO mode only. Used to input MSDC write data to card. The access can be AHB 1B/2B/4B
112c001C	**MSDC_RXDATA**	32	**MSDC RX Data Port Register** The register is for PIO mode only. Used to read back MSDC read data from card. The access can be AHB 1B/2B/4B.
112c0030	**SDC_CFG**	32	**SD Configuration Register** The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
112c0034	**SDC_CMD**	32	**SD Command Register** The register defines a SD Memory Card command and its attributes. Before MS/SD controller issues a transaction onto SD

			bus, application shall specify other relative settings such as argument for command. After writing the register by the application, MS/SD controller will issue the corresponding transaction onto SD serial bus. If the command is GO_IDLE_STATE, the controller will have serial clock on SD/MMC bus run 128 cycles before issuing the command.
112c0038	**SDC_ARG**	32	**SD Argument Register** The register contains the argument of the SD/MMC Memory Card command.
112c003C	**SDC_STS**	32	**SD Status Register** The register reflects SD bus status and contains MMC stream write status.
112c0040	**SDC_RESP0**	32	**SD Response Register 0** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
112c0044	**SDC_RESP1**	32	**SD Response Register 1** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
112c0048	**SDC_RESP2**	32	**SD Response Register 2** The register contains parts of the last SD/MMC Memory Card bus response. See description for the register field SDC_RESP3.
112c004C	**SDC_RESP3**	32	**SD Response Register 3** The register contains parts of the last SD/MMC Memory Card bus response. The register fields SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3 are composed of the last SD/MMC Memory card bus response. For response of type R2, that is, response of the command ALL_SEND_CID, SEND_CSD and SEND_CID, only bit 127 to 0 of response token is stored in the register field SDC_RESP0, SDC_RESP1, SDC_RESP2 and SDC_RESP3. SDC_RESP0 = bit 31~0 SDC_RESP1 = bit 63~32 SDC_RESP2 = bit 95~64 SDC_RESP3 = bit 127~96 For response of type R1b in auto CMD12 or R1 in auto CMD23, bit 39 to 8 of response token is stored in the register field of SDC_RESP3. For the responses of other types, only bit 39 to 8 of response token is stored in the register field SDC_RESP0.
112c0050	**SDC_BLK_NUM**	32	**SD Block Number Register** This register defines the block number for the block transaction. For single read/write, this register should be set to 1. For multiple read/write, this register should be set to larger than 1. Set to 0 will cause unexpected result.
112c0054	**SDC_VOL_CHG**	32	**SD Voltage Change Wait Time Register** This register define SD voltage change check wait time
112c0058	**SDC_CSTS**	32	**SD Card Status Register** After commands with R1 and R1b response, this register will contain the status of the SD/MMC card
112c005C	**SDC_CSTS_EN**	32	**SD Card Status Enable Register** This register is used to control which bit of the SDC_CSTS will generate the MSDC_INT.SD_CSTA interrupt.
112c0060	**SDC_DATCRC_STS**	32	**SD Card Data CRC Status Register** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read
112c0070	**EMMC_CFG0**	32	**EMMC Configuration Register 0** The register is used for boot up mode general configuration of e-MMC version 4.3 and 4.4.
112c0074	**EMMC_CFG1**	32	**EMMC Configuration Register 1** The register is used for boot up mode general configuration of e-

			MMC version 4.3 and 4.4.
112c0078	EMMC_STS	32	**EMMC Status Register** The register reflects the status of e-MMC boot up mode operation.
112c007C	EMMC_IOCON	32	**EMMC IO Control Register** The register controls the H/W reset pin of e-MMC boot up mode operation.
112c0080	SD_ACMD_RESP	32	**SD ACMD Response Register** This register stores the response of auto command from SD card
112c0084	SD_ACMD19_TRG	32	**SD ACMD19 Target Register** This register is used to select target delay line to run ACMD19 sequence.
112c0088	SD_ACMD19_STS	32	**SD ACMD19 Status Register** This register stores the result of auto command 19 from SD card
112c008C	DMA_SA_HIGH4BIT	32	**DMA Current Address Resgiter of high 4bit** This register contain the start address high 4bit of 36bit address for 64G dram access
112c0090	DMA_SA	32	**DMA Start Address Register** This register contains the start address of the DMA descriptor
112c0094	DMA_CA	32	**DMA Current Address Register** This register contains the current DMA address
112c0098	DMA_CTRL	32	**DMA Control Register** This register is used to control the DMA operation.
112c009C	DMA_CFG	32	**DMA Configuration Register** This register is used to configure the DMA operation.
112c00A0	SW_DBG_SEL	32	**MSDC S/W Debug Selection Register** This register is used to select S/W debug output
112c00A4	SW_DBG_OUT	32	**MSDC S/W Debug Output Register** This register shows the selected debug output
112c00A8	DMA_LENGTH	32	**DMA Length Register** This register is used to set Basic DMA operation length
112c00B0	PATCH_BIT0	32	**MSDC Patch Bit Register 0** This register can configure the patch function. For normal function, these bit should keep in default value
112c00B4	PATCH_BIT1	32	**MSDC Patch Bit Register 1** This register can configure the patch function. For normal function, these bit should keep in default value
112c00B8	PATCH_BIT2	32	**MSDC Patch Bit Register 2** This register can configure the patch function. For normal function, these bit should keep in default value
112c00F0	PAD_TUNE0	32	**MSDC Pad Tuning Register0** This register can configure the delay line embedded in Pad Macro
112c00F4	PAD_TUNE1	32	**MSDC Pad Tuning Register1** This register can configure the delay line embedded in Pad Macro
112c00F8	DAT_RD_DLY0	32	**MSDC Data Delay Line Register 0** This register can configure the delay line embedded in Pad Macro
112c00FC	DAT_RD_DLY1	32	**MSDC Data Delay Line Register 1** This register can configure the delay line embedded in Pad Macro
112c0100	DAT_RD_DLY2	32	**MSDC Data Delay Line Register 2** This register can configure the delay line embedded in Pad Macro
112c0104	DAT_RD_DLY3	32	**MSDC Data Delay Line Register 3** This register can configure the delay line embedded in Pad Macro
112c0110	HW_DBG_SEL	32	**MSDC H/W Debug Selection Register** This register can select the H/W debug output

112c0114	MAIN_VER	32	**MSDC Main Version Register** This register shows the version code of MSDC IP
112c0118	ECO_VER	32	**MSDC ECO Version Register** This register shows the ECO version code of MSDC IP
112c0180	EMMC50_PAD_CTL0	32	**MSDC eMMC5.0 CLK PAD Control Register 0** This register can configure the eMMC5.0 SD CLK pad for duty cycle corrector
112c0184	EMMC50_PAD_DS_CTL0	32	**MSDC eMMC5.0 DS PAD Control Register 0** This register can configure the eMMC5.0 SD DS pad
112c0188	EMMC50_PAD_DS_TUNE	32	**MSDC eMMC5.0 DS Pad Tuning Register** This register can configure the delay line embedded in DS Pad Macro
112c018C	EMMC50_PAD_CMD_TUNE	32	**MSDC eMMC5.0 CMD Pad Tuning Register**
112c0190	EMMC50_PAD_DAT01_TUNE	32	**MSDC eMMC5.0 DAT0/1 Pad Tuning Register**
112c0194	EMMC50_PAD_DAT23_TUNE	32	**MSDC eMMC5.0 DAT2/3 Pad Tuning Register**
112c0198	EMMC50_PAD_DAT45_TUNE	32	**MSDC eMMC5.0 DAT4/5 Pad Tuning Register**
112c019C	EMMC50_PAD_DAT67_TUNE	32	**MSDC eMMC5.0 DAT6/7 Pad Tuning Register**
112c0204	EMMC51_CFG0	32	**eMMC51 configuration** register0 The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
112c0208	EMMC50_CFG0	32	**eMMC50 configuration** register0 The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
112c020C	EMMC50_CFG1	32	**eMMC50 configuration** register1 The register is used for configuring the MS/SD Memory Card Controller when it is configured as the host of SD Memory Card. If the controller is configured as the host of Memory Stick, the contents of the register have no impact on the operation of the controller.
112c021C	EMMC50_CFG2	32	**ahb2axi wrapper control register** this register can set wrapper work mode
112c0220	EMMC50_CFG3	32	
112c0224	EMMC50_CFG4	32	
112c0228	EMMC50_BLOCK_LENGTH	32	

112c0000 MSDC_CFG MSDC Configuration Register 02000099

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							SCLKSTOPSEL	START_BIT		HS400CKMD	CCKMD		CCKDIV[11:8]			
Type							RW	RW		RW	W1C		RW			

Reset							1	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCKDIV[7:0]								CCKSB	BV18PSS	BV18SDT	CCKDRVE	PIO	RST	CCKPD	MSDC
Type	RW								RU	RU	RW	RW	RW	A0	RW	RW
Reset	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1

Bit(s)	Mnemonic	Name	Description
25	**SCLKSTOPSEL**	SCLK_STOP_SEL	**In DDR mode,stop SCLK when device is idle whether check SCLK phase** 1'b0: stop sclk no check SCLK phase 1'b1: stop sclk check SCLK phase,fix 1/4T sclk glitch in DDR mode
24:23	**START_BIT**	START_BIT	**Check read data start bit slection in DDR mode** 2'b00: rising edge 2'b01: falling edge 2'b10: rising & falling edge 2'b11: rising | falling edge
22	**HS400CKMD**	HS400_CK_MODE	**HS400 clock mode selection** 1'b0: HS400 source clock is 800M 1'b1: HS400 source clock is 400M
21:20	**CCKMD**	CARD_CK_MODE	**MS/SD Card clock mode** 2'b00: Use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b01: Use msdc_src_ck as msdc_ck, bit[15]~bit[8] is ignored. 2'b10: DDR mode, also use clock divider output which divided by msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed. 2'b11: HS400 mode,also use clock divider output and use msdc_src_ck as msdc_ck, bit[15]~bit[8] should be programmed.
19:8	**CCKDIV**	CARD_CK_DIV	**MS/SD Card clock divider** The register field controls clock frequency of serial clock on MS/SD bus. Please refer to Data Line Latching Timing Diagram and Response Latching Timing Diagram. For non-DDR mode, msdc_ck equals SD bus clock. (Ex: For SDR25 or HS, msdc_ck and SD bus clock will be 50MHz) For DDR mode, msdc_ck denotes the MSDC internal clock which will be dobule to SD bus clock. (Ex: For DDR50, msdc_ck should be set to 100MHz and bus clock will be 50MHz) 8'b00000000: msdc_ck =(1/2) * msdc_src_ck 8'b00000001: msdc_ck = (1/(4*1)) * msdc_src_ck 8'b00000010: msdc_ck = (1/(4*2)) * msdc_src_ck 8'b00000011: msdc_ck = (1/(4*3))* msdc_src_ck 8'b00010000: msdc_ck = (1/(4*16))* msdc_src_ck 8'b11111111: msdc_ck = (1/(4*255)) * msdc_src_ck
7	**CCKSB**	CARD_CK_STABLE	**MS/SD Card clock stable or not** After programming the CARD_CK_MODE or CARD_CK_DIV, this bit will immediately go to "0" and return to "1" if stable. User should poll this register to make sure the safety control of MSDC. 1'b0: Clock output is not stable 1'b1: Clock output is stable
6	**BV18PSS**	BV_1P8_PASS	**MSDC Bus voltage 1.8V detection status** S/W should check this bit after BUS_VOL_18V_START_DET turns to 0 from 1. 1'b0: The voltage detection has error. 1'b1: The voltage detection has no error.
5	**BV18SDT**	BV_1P8_START_DET	**MSDC Bus voltage 1.8V detection sequence start event** S/W writes this bit to 1 to trigger H/W outputs 1.8V clock for 1 ms and automatically detect CMD/DAT line sequence for voltage change is passed or not. H/Q will clear this bit to 0 after the detection has finished.

Bit(s)	Mnemonic	Name	Description
			The pass or fail status is stored in bit[6] BUS_VOL_18V_PASS.
4	**CCKDRVE**	CARD_CK_DRV_EN	**SD/MS Card Bus Clock drive enable bit** Set this bit to 1 to enable MSDC bus clock driver. The default bus state depends on MSDC_CFG][1] CARD_CK_PWDN bit. If MSDC_CFG[1] CARD_CK_PWDN= 1, the default clock state is free running. If MSDC_CFG[1] CARD_CK_PWDN = 0, the default clock state is gated to 0. Set this bit to 0 will put the bus state into "tri-state". Default is 1. 1'b0: Put the clock pad into tri-state 1'b1: Enable MSDC to drive clock pad, the state of CLK depends on MSDC_CFG[1] CARD_CK_PWDN
3	**PIO**	PIO_MODE	**MS/SD PIO mode** PIO mode selection. Default is in PIO mode. 1'b0: DMA mode 1'b1: PIO mode
2	**RST**	RST	**Software reset** Writing 1 to this register will cause internal synchronous reset of MS/SD controller, and it will not reset register settings and DMA controller. The reset sequence is done when this bit goes to 0. S/W should wait this bit back to 0 after writing 1. 1'b0: MS/SD controller is not in reset state 1'b1: MS/SD controller is in reset state
1	**CCKPD**	CARD_CK_PWDN	**MSDC bus clock power down mode** This bit controls the card clock power down mode. 1'b0: Clock is gated to 0 if no command or data is transmitted. 1'b1: Clock is free running even if no command or data is transmitted. (The clock may still be stopped when MSDC write data is not enough or no space for next read data)
0	**MSDC**	MSDC	**MS/SD mode selection** The register bit is used to configure the controller as the host of Memory Stick or as the host of SD/MMC Memory card. The default value is to configure the controller as the host of Memory Stick. 1'b0: Configure the controller as the host of Memory Stick 1'b1: Configure the controller as the host of SD/MMC Memory card

112c0004 MSDC_IOCON MSDC IO Configuration Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									RD7SPL	RD6SPL	RD5SPL	RD4SPL	RD3SPL	RD2SPL	RD1SPL	RD0SPL
Type									RW	RW	RW	RW	RW	RW	RW	RW
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			WD3SPL	WD2SPL	WD1SPL	WD0SPL	WDSPLSEL	WDSPL			RDSPLSEL	DDR50CKD	DDLSEL	RDSPL	RSPL	SDR104CKS
Type			RW	RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23	**RD7SPL**	R_D7_SMPL	**Read data 7 sample selection** This bit is only valid when bit 5 is ON

Bit(s)	Mnemonic	Name	Description
			1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
22	**RD6SPL**	R_D6_SMPL	**Read data 6 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
21	**RD5SPL**	R_D5_SMPL	**Read data 5 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
20	**RD4SPL**	R_D4_SMPL	**Read data 4 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
19	**RD3SPL**	R_D3_SMPL	**Read data 3 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
18	**RD2SPL**	R_D2_SMPL	**Read data 2 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
17	**RD1SPL**	R_D1_SMPL	**Read data 1 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
16	**RD0SPL**	R_D0_SMPL	**Read data 0 sample selection** This bit is only valid when bit 5 is ON 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
13	**WD3SPL**	W_D3_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
12	**WD2SPL**	W_D2_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
11	**WD1SPL**	W_D1_SMPL	**SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample SDIO interrupt by external bus clock rising edge 1'b1: Sample SDIO interrupt by external bus clock falling edge
10	**WD0SPL**	W_D0_SMPL	**CRC Status and SDIO interrupt sample selection** This bit is only valid when bit 9 is ON 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
9	**WDSPLSEL**	W_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in write transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.W_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.W_D0_SMPL Data line 1: MSDC_IOCON.W_D1_SMPL

Bit(s)	Mnemonic	Name	Description
			Data line 2: MSDC_IOCON.W_D2_SMPL Data line 3: MSDC_IOCON.W_D3_SMPL
8	**WDSPL**	W_D_SMPL	**CRC Status and SDIO interrupt sample selection** 1'b0: Sample CRC Status and SDIO interrupt by external bus clock rising edge 1'b1: Sample CRC Status and SDIO interrupt by external bus clock falling edge
5	**RDSPLSEL**	R_D_SMPL_SEL	**Data line rising/falling latch fine tune selection in read transaction** 1'b0: All data line share one value indicated by MSDC_IOCON.R_D_SMPL 1'b1: Each data line has its own selection value indicated by Data line 0: MSDC_IOCON.R_D0_SMPL Data line 1: MSDC_IOCON.R_D1_SMPL Data line 2: MSDC_IOCON.R_D2_SMPL Data line 3: MSDC_IOCON.R_D3_SMPL Data line 4: MSDC_IOCON.R_D4_SMPL Data line 5: MSDC_IOCON.R_D5_SMPL Data line 6: MSDC_IOCON.R_D6_SMPL Data line 7: MSDC_IOCON.R_D7_SMPL
4	**DDR50CKD**	DDR50_DLY_SEL	**DDR50 output clock delay selection** 1'b0: Use default clock output 1'b1: Delay 1T msdc_src_ck for clock output
3	**DDLSEL**	D_DLYLINE_SEL	**Data line delay line fine tune selection** 1'b0: All data line share one delay selection value indicated by PAD_TUNE.PAD_DAT_RD_RXDLY 1'b1: Each data line has its own delay selection value indicated by Data line 0: DAT_RD_DLY0.DAT0_RD_DLY Data line 1: DAT_RD_DLY0.DAT1_RD_DLY Data line 2: DAT_RD_DLY0.DAT2_RD_DLY Data line 3: DAT_RD_DLY0.DAT3_RD_DLY Data line 4: DAT_RD_DLY1.DAT4_RD_DLY Data line 5: DAT_RD_DLY1.DAT5_RD_DLY Data line 6: DAT_RD_DLY1.DAT6_RD_DLY Data line 7: DAT_RD_DLY1.DAT7_RD_DLY
2	**RDSPL**	R_D_SMPL	**Read data sample selection** 1'b0: Sample read data by external bus clock rising edge 1'b1: Sample read data by external bus clock falling edge
1	**RSPL**	R_SMPL	**Command response sample selection** 1'b0: Sample response by external bus clock rising edge 1'b1: Sample response by external bus clock falling edge
0	**SDR104CKS**	SDR104_CLK_SEL	**SDR104 SCLK output clock control** This bit is only used when MSDC_CFG[17:16] CARD_CK_MODE is 2'b01. 1'b0: Bus clock output equals inverted msdc_src_ck 1'b1: Bus clock output equals msdc_src_ck

112c0008 MSDC_PS MSDC Pin Status Register 01FF0002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SDWP							CMD	DAT							
Type	RU							RU	RU							
Reset	0							1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CDDBCE														CDST	CDEN

														S	
Type	RW													RU	RW
Reset	0	0	0	0										1	0

Bit(s)	Mnemonic	Name	Description
31	**SDWP**	SD_WP	**Write Protection Switch status on SD Memory Card** The register bit shows the status of Write Protection Switch on SD Memory Card. There is no default reset value. The pin WP (Write Protection) is only useful while the controller is configured for SD Memory Card 1'b0: Write Protection Switch ON. It means that memory card is desired to be write-protected 1'b1: Write Protection Switch OFF. It means that memory card is writable
24	**CMD**	CMD	**Command line status** This bit reflects the command line value of MSDC bus.
23:16	**DAT**	DAT	**Data line status** This bit reflects the data line value of MSDC bus. (8-bits)
15:12	**CDDBCE**	CDDEBOUNCE	**Card detection de-bounce timer** The register field specifies the time interval for card detection de-bounce. Its default value is 0. It means that de-bounce interval is one 32KHz cycle. The interval will extend one cycle time of 32KHz by increasing the counter by 1
1	**CDSTS**	CDSTS	**Card detection status** 1'b1: Card detection pin status is logic high 1'b1: Card detection pin status is logic high
0	**CDEN**	CDEN	**Card detect enable** The register bit is used to control the card detection circuit 1'b0: Card detection is disable 1'b1: Card detection is enable

112c000C MSDC_INT MSDC Interrupt Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									AXIRSPERR				DMAPROTECT	GPDCSERR	BDCSERR	AC19DONE
Type									W1C				W1C	W1C	W1C	W1C
Reset									0				0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SDDCRCERR	SDDTO	DMAXFDNE	SDXFCPL	SDCSTS	SDRCRCER	SDCTO	SDCRDY		DMAQEPTY	SDACDRCRCER	SDACDCTO	SDACDCRDY		MSDCCDSC	MMCIRQ
Type	W1C	W1C	W1C	W1C	RU	W1C	W1C	W1C		W1C	W1C	W1C	W1C		W1C	W1C
Reset	0	0	0	0	0	0	0	0		0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
23	**AXIRSPERR**	AXI_RESP_ERR	**AXI BUS response error status**
19	**DMAPROTECT**	DMA_PROTECT	**there is write operation to DMA start address, length, start bit or last buf bit**
18	**GPDCSERR**	GPD_CS_ERR	**GPD checksum error detected**
17	**BDCSERR**	BD_CS_ERR	**BD checksum error detected**
16	**AC19DONE**	AUTOCMD19_DONE	**SD Auto command 19 status register** When auto-command 19 is enabled, H/W will clear this register to 0.

Bit(s)	Mnemonic	Name	Description
			As the tuning sequence is finished (32 times), this register will be set to 1 by H/W. S/W should check the AUTOCMD_STS0 and SUTOCMD_STS1 only when this bit is ON.
15	**SDDCRCERR**	SD_DATA_CRCERR	**SD Data CRC error interrupt** Indicates that MS/SD controller detects a CRC error after reading a block of data from the DAT line or SD/MMC signals a CRC error after writing a block of data to the DAT line. 1'b0: Otherwise 1'b1: MS/SD controller detected a CRC error after reading a block of data from the DAT line or SD/MMC signaled a CRC error after writing a block of data to the DAT line
14	**SDDTO**	SD_DATTO	**SD Data timeout interrupt** Indicates that SD/MMC controller detects a timeout condition while waiting for data token on the DAT line. This bit is for both data read and data write. For SD data read, timeout will occur when the read data is not presented. For SD data write, timeout will occur when the write data CRC status is not presented if PATCH_BIT[30] DETECT_WR_CRC_TIMEOUT = 1 1'b0: Otherwise 1'b1: SD/MMC controller detects a timeout condition while waiting for data token on the DAT line
13	**DMAXFDNE**	DMA_XFER_DONE	**DMA transfer done interrupt** The register bit indicates the status of data block transfer. 1'b0: Otherwise 1'b1: A data block was successfully transferred
12	**SDXFCPL**	SD_XFER_COMPLETE	**SD Data transfer complete interrupt** This bit indicates the transaction which contains data has completed. While performing tuning procedure (Execute Tuning is set to 1), SD_XFER_COMPLETE is not set to 1.
11	**SDCSTS**	SD_CSTS	**SD CSTA update interrupt** The register bit indicates any bit in the register SDC_CSTA is active, the register bit will be set to 1. S/W should clear the SDC_CSTA and this bit will be de-asserted automatically. 1'b0: No SD Memory Card interrupt 1'b1: SD Memory Card interrupt exists
10	**SDRCRCER**	SD_RESP_CRCERR	**SD Command CRC error interrupt** Indicates that SD/MMC controller detected a CRC error after reading a response from the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a CRC error after reading a response from the CMD line
9	**SDCTO**	SD_CMDTO	**SD Command timeout interrupt** Indicates that SD/MMC controller detected a timeout condition while waiting for a response on the CMD line. 1'b0: Otherwise 1'b1: SD/MMC controller detected a timeout condition while waiting for a response on the CMD line
8	**SDCRDY**	SD_CMDRDY	**SD Command ready interrupt** For the command without response, the register bit will be 1 once the command completes on SD/MMC bus. For command with response without busy, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error.

Bit(s)	Mnemonic	Name	Description
			For command with response with busy in DAT0, the register bit will be 1 whenever the command is issued onto SD/MMC bus and its corresponding response is received without CRC error and the DAT0 transited from busy to idle. 1'b0: Otherwise 1'b1: Command finish successfully without a CRC error
6	**DMAQEPTY**	DMA_Q_EMPTY	**DMA queue empty interrupt** This bit is used to indicate the current DMA queue is empty. Only for Descriptor mode and Enhance mode.
5	**SDACDRCRCER**	SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt** This bit is set when detecting a CRC error in the Auto command response.
4	**SDACDCTO**	SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt** This bit is set if no response is returned within a specified cycles(64T in spec) from the end bit of Auto command.
3	**SDACDCRDY**	SD_AUTOCMD_CMDRDY	**SD auto command ready interrupt** This bit is set if auto command is executed without CRC error or time out.
1	**MSDCCDSC**	MSDC_CDSC	**MSDC Card detection status change interrupt** The register bit indicates if any interrupt for memory card insertion/removal exists. Whenever memory card is inserted or removed and card detection circuit is enabled, i.e., the register bit CDEN in the register MSDC_PS is set to 1, the register bit will be set to 1. It will be reset when the register is read. 1'b0: Otherwise 1'b1: Card is inserted or removed
0	**MMCIRQ**	MMC_IRQ	**MMC card interrupt** 1'b0: Otherwise 1'b1: indicates that MMC card interrupt event occurs

112c0010 MSDC_INTEN MSDC Interrupt Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									ENAXIRSPERR				ENDMAPROTECT	ENGPDCSERR	ENBDCSERR	ENAC19DONE
Type									RW				RW	RW	RW	RW
Reset									0				0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENSDDCRCERR	ENSDDTO	ENDMAXFDNE	ENSDXFCPL	ENSDCSTA	ENSDRCRCER	ENSDCTO	ENSDCRDY	ENSDIOIRQ	ENDMAQEPTY	ENSDACDRCRCER	ENSDACDCTO	ENSDACDCRDY		ENMSDCCDSC	ENMMCIRQ
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0

Bit(s)	Mnemonic	Name	Description
23	**ENAXIRSPERR**	EN_AXI_RESP_ERR	**AXI BUS response error status interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
19	**ENDMAPROTECT**	EN_DMA_PROTECT	**DMA protection interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
18	**ENGPDCSE**	EN_GPD_CS_ERR	**GPD checksum error interrupt enable**

Bit(s)	Mnemonic	Name	Description
	RR		1'b0: Disable interrupt 1'b1: Enable interrupt
17	**ENBDCSERR**	EN_BD_CS_ERR	**BD checksum error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
16	**ENAC19DONE**	EN_AUTOCMD19_DONE	**Auto-command 19 complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
15	**ENSDDCRCERR**	EN_SD_DATA_CRCERR	**SD Data CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
14	**ENSDDTO**	EN_SD_DATTO	**SD Data timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
13	**ENDMAXFDNE**	EN_SD_DMA_XFER_DONE	**DMA transfer done interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
12	**ENSDXFCPL**	EN_SD_XFER_COMPLETE	**SD Data transfer complete interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
11	**ENSDCSTA**	EN_SD_CSTA	**SD CSTA update interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
10	**ENSDRCRCER**	EN_SD_RESP_CRCERR	**SD Command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
9	**ENSDCTO**	EN_SD_CMDTO	**SD Command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
8	**ENSDCRDY**	EN_SD_CMDRDY	**SD Command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
7	**ENSDIOIRQ**	EN_SD_SDIOIRQ	**SD SDIO interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
6	**ENDMAQEPTY**	EN_DMA_Q_EMPTY	**DMA queue empty interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
5	**ENSDACDRCRCER**	EN_SD_AUTOCMD_RESP_CRCERR	**SD auto command CRC error interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
4	**ENSDACDCTO**	EN_SD_AUTOCMD_CMDTO	**SD auto command timeout interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
3	**ENSDACDCRDY**	EN_AUTOCMD_CMDRDY	**SD auto command ready interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
1	**ENMSDCCDSC**	EN_MSDC_CDSC	**MSDC Card detection status change interrupt enable** 1'b0: Disable interrupt 1'b1: Enable interrupt
0	**ENMMCIRQ**	EN_MMC_IRQ	**MMC card interrupt enable** 1'b0: Disable interrupt

Bit(s)	Mnemonic	Name	Description
			1'b1: Enable interrupt

112c0014 MSDC_FIFOCS MSDC FIFO Control and Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFOCLR								TXFIFOCNT							
Type	A0								RU							
Reset	0								0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									RXFIFOCNT							
Type									RU							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31	**FIFOCLR**	FIFOCLR	**Embedded FIFO clear** Write this bit to 1 makes FIFO cleared. It will goes to 0 when FIFO is cleared. S/W needs to check this bit to make sure clearing FIFO sequence is done. This bit can be used when the data read/write sequence has error and need to clean the H/W FIFO.
23:16	**TXFIFOCNT**	TXFIFOCNT	**TX FIFO count for MSDC write** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved
7:0	**RXFIFOCNT**	RXFIFOCNT	**RX FIFO count for MSDC read** 8'd0: No data in FIFO 8'd1: 1bytes data in FIFO 8'd2: 2 bytes data in FIFO 8'd131: Maximum 131 bytes data in FIFO Others: reserved

112c0018 MSDC_TXDATA MSDC TX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIOTXDATA[31:16]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIOTXDATA[15:0]															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIOTXDATA**	PIO_TXDATA	**PIO mode TXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

112c001C MSDC_RXDATA MSDC RX Data Port Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIORXDATA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIORXDATA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**PIORXDATA**	PIO_RXDATA	**PIO mode RXDATA port** This register can be accessed by Byte or Half-word or Word. This port can only be accessed in PIO mode. Otherwise, the transaction will be discarded.

112c0030 SDC_CFG SD Configuration Register 00100000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DTOC										INTBGP	SDIOIDE	SDIO		BUSWD	
Type	RW										RW	RW	RW		RW	
Reset	0	0	0	0	0	0	0	0			0	1	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															ENWKUPINS	ENWKUPSDIOINT
Type															RW	RW
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31:24	**DTOC**	DTOC	**Data Timeout Counter** The period from the end of the initial host read command or the last read data block in a multiple block read operation to the start bit of the next read data block requires at least two serial clock cycles. The counter is used to extend the period (Read Data Access Time) in unit of 1048576 serial clocks. 8'b00000000: Extend 1048576 more serial clock cycle 8'b00000001: Extend 1048576x2 more serial clock cycle 8'b00000010: Extend 1048576x3 more serial clock cycle 8'b11111111: Extend 1048576x 256 more serial clock cycle
21	**INTBGP**	INT_AT_BLOCK_GAP	**Interrupt at block Gap** This bit is valid only in 4-bit mode of the SDIO card and selects a sample point in the interrupt cycle. Setting to 1 enables interrupt detection at the block gap for a multiple block transfer. Setting to 0 disables interrupt detection during a multiple block transfer. If the SD card cannot signal an interrupt during a multiple block transfer, this bit should be set to 0. When the Host Driver detects an SD card insertion, it shall set this bit according to the CCCR of the SDIO card. 1'b0: Disables interrupt detection at the block gap 1'b1: Enables interrupt detection at the block gap
20	**SDIOIDE**	SDIO_INT_DET_EN	**SDIO interrupt detection enable** This bit is to inform the SD controller to sense the SDIO interrupt 1'b0: SDIO interrupt detection is disabled 1'b1: SDIO interrupt detection is enabled if the SDIO bit is also on
19	**SDIO**	SDIO	**SDIO mode enable bit**

Bit(s)	Mnemonic	Name	Description
			This bit is to enable the support to sense the SDIO interrupt and disable the R4 response CRC check for SDIO card 1'b0: SDIO mode is disabled 1'b1: SDIO mode is enabled
17:16	**BUSWD**	BUSWIDTH	**Bus width configuration** This field is used to define the SD/MMC bus width 2'b00: 1 bit mode 2'b01: 4 bit mode 2'b10: 8 bit mode 2'b11: reserved
1	**ENWKUPINS**	WAKEUP_INS_EN	**Card status change wakeup event enable bit** 1'b0: Disable wakeup event for card status change 1'b1: Enable wakeup event for card status change
0	**ENWKUPSDIOINT**	WAKEUP_SDIOINT_EN	**SDIO card interrupt wakeup event enable bit** 1'b0: Disable wakeup event for SDIO card interrupt 1'b1: Enable wakeup event for SDIO card interrupt

112c0034 SDC_CMD SD Command Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		VOLSWTH	ACMD		LEN											
Type		RW	RW		RW											
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GOIRQ	STOP	RW	DTYPE			RSPTYP			BREAK	CMD					
Type	RW	RW	RW	RW			RW			RW	RW					
Reset	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30	**VOLSWTH**	VOL_SWTH	**Voltage switch command** 1'b0: Disable voltage switch detection 1'b1: Enable voltage switch detection
29:28	**ACMD**	AUTO_CMD	**Auto command enable** This field determines use of auto command functions. This function can be used in all modes including PIO/Basic DMA/Descriptor DMA/Enhanced Mode. There are two methods to stop Multiple-block read and write operation. (1) Auto CMD12 Enable Multiple-block read and write commands for memory require CMD12 to stop the operation. When ACMD-12 is used, MSDC issues CMD12 automatically when last block transfer is completed. Auto CMD12 error is indicated to the MSDC_INT register. The Host Driver shall not set this bit if the command does not require CMD12. In particular, secure commands defined in the Part 3 File Security specification do not require CMD12. (2) Auto CMD23 Enable When ACMD-23 is used, MSDC issues a CMD23 automatically before issuing a command specified in the CMD field. The Host Controller Version 3.00 and later shall support this function. By writing the Command register, MSDC issues a CMD23 first and then issues a command specified by the CMD field in SDC_CMD register. If response errors of CMD23 are detected, the second command is not issued. A CMD23 error is indicated in the MSDC_INT register.

Bit(s)	Mnemonic	Name	Description
			32-bit block count value for CMD23 is set to SDC_BLOCK_NUM register. 2'b00: Disable Auto Command 2'b01: Enable Auto CMD12 2'b10: Enable Auto CMD23 2'b11: Enable Re-tuning CMD19
27:16	**LEN**	LEN	**Length** The register field is used to define the length of one block in unit of byte in a data transaction of block mode or the data length in unit of byte in data transaction of byte mode. The maximal value of block length is 2048 bytes. 12'b000000000000: Reserved 12'b000000000001: Block length is 1 byte 12'b000000000010: Block length is 2 byte 12'b011111111111: Block length is 2047 byte 12'b100000000000: Block length is 2048 byte
15	**GOIRQ**	GO_IRQ	**GO_IRQ command** The register bit indicates if the command is GO_IRQ_STATE (CMD40) and used only for MMC protocol. If the command is GO_IRQ_STATE, the period between command token and response token will not be limited. 1'b0: The command is not GO_IRQ_STATE 1'b1: The command is GO_IRQ_STATE
14	**STOP**	STOP	**Stop command** The register bit indicates if the command is a stop transmission command. It should be set to 1 when CMD12 (SD/MMC) or CMD52 with I/O abort (SDIO) is to be issued. 1'b0: The command is not a stop transmission command 1'b1: The command is a stop transmission command
13	**RW**	RW	**Command read write selection** The register bit defines the command is a read command or write command. The register bit is valid only when the command will cause a transaction with data token. 1'b0: The command is a read command 1'b1: The command is a write command
12:11	**DTYPE**	DTYPE	**Data block selection** The register field defines data token type for the command. 2'b00: No data token for the command 2'b01: Single block transaction (only available in block mode) 2'b10: Multiple block transaction. (only available in block mode) 2'b11: Stream operation. It only shall be used in MMC protocol. (only available in block mode)
9:7	**RSPTYP**	RSPTYP	**Command response type** 3'b000: This command has no response. 3'b001: The command has R1/R5/R6/R7 response. The response token is 48-bit with CRC check (For SD/MMC/SDIO) (Not include the SDIO abort command) 3'b010: The command has R2 response. The response token is 136-bit (For SD/MMC) 3'b011: The command has R3 response. The response token is 48-bit response, no CRC check (For SD/MMC) 3'b100: The command has R4 response. The response token is 48-bit without CRC check (For SDIO) The response token is 48-bit with CRC check (For MMC) 3'b111: The command has R1b response. The response token is 48-bit (For SD/MMC/SDIO)
6	**BREAK**	BREAK	**Abort a pending MMC GO_IRQ command** It is only valid for a pending GO_IRQ_MODE command waiting for

Bit(s)	Mnemonic	Name	Description
			MMC interrupt response. 1'b0: Not a beak command 1'b1: Break a pending MMC GO_IRQ_MODE command in the controller.
5:0	**CMD**	CMD	**SD Memory Card command**

112c0038 SDC_ARG SD Argument Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ARG[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ARG[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ARG**	ARG	**Memory card controller argument register**

112c003C SDC_STS SD Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MMCSWRCPL															CMD_WR_BUSY
Type	RU															W1C
Reset	0															0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															CMDBSY	SDCBSY
Type															RU	RU
Reset															0	0

Bit(s)	Mnemonic	Name	Description
31	**MMCSWRCPL**	MMC_STREAM_WR_COMPL	**MMC Stream mode write data is all flushed to MMC card** S/W can use this bit to confirm last write data are flushed to MMC then issue STOP command. This bit is only valid when the command SDC_CMD.DTYPE=2'b11. 1'b0: Last Data are partially inside MSDC 1'b1: Last data are flushed to MMC card
16		CMD_WR_BUSY	
1	**CMDBSY**	CMDBUSY	**SD Command line busy status** S/W should always read this bit to make sure the command line is not busy before sending the next command. If the command is R1B or data read/write command, S/W should check SDCBUSY bit too. Note: When Auto command 12 is enabled, this bit will be asserted immediately after SDC_CMD is written and de-asserted after auto-command 12 finishes. 1'b0: No transmission is going on CMD line on SD bus 1'b1: There exists transmission going on CMD line on SD bus
0	**SDCBSY**	SDCBUSY	**SD controller busy status**

Bit(s)	Mnemonic	Name	Description
			1'b0: SD controller is idle 1'b1: SD controller is busy

112c0040 SDC_RESP0 SD Response Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP0[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP0[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP0**	RESP0	**Memory card controller response register 0**

112c0044 SDC_RESP1 SD Response Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP1[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP1[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP1**	RESP1	**Memory card controller response register 1**

112c0048 SDC_RESP2 SD Response Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP2[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP2[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP2**	RESP2	**Memory card controller response register 2**

112c004C SDC_RESP3 SD Response Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESP3[31:16]															

Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESP3[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**RESP3**	RESP3	**Memory card controller response register 3**

112c0050 SDC_BLK_NUM SD Block Number Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BLKNUM[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BLKNUM[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:0	**BLKNUM**	BLOCK_NUMBER	**Memory card controller Block number** This field indicates the block number of data transaction. 32'd0: Reserved 32'd1: 1 data block 32'd2: 2 data block 32'd3: 3 data block 32'hffffffff: 4GB-1 data block

112c0054 SDC_VOL_CHG SD Voltage Change Wait Time Register 00000145

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VCHGCNT															
Type	RW															
Reset	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
15:0	**VCHGCNT**	VOL_CHG_WAIT_CNT	**This register define SD voltage change check wait time,wait time is clock frequency multiply VOL_WAIT_TIME**

112c0058 SDC_CSTS SD Card Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS[31:16]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	CSTS[15:0]															
Type	W1C															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS**	CSTS	**CSTS** The card status field in the response R1 or R1b field. Each bit can be write 1 clear individually.

112c005C SDC_CSTS_EN SD Card Status Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CSTS_EN[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CSTS_EN[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**CSTS_EN**	CSTS_EN	**CSTS_EN** This register is used to control which bit of the CSTA will generate the MSDC_INT.SDCSTA

112c0060 SDC_DATCRC_STS SD Card Data CRC Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCSSN								DCSSP							
Type	RU								RU							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8	**DCSSN**	DAT_CRCSTS_NEG	**MSDC DDR mode negative edge Read DATA CRC status** This register reflects the CRC status of data line[7:0] in DDR mode. The positive edge CRC status is shown in DAT_CRC_STS[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error
7:0	**DCSSP**	DAT_CRCSTS_POS	**MSDC read DATA CRC status** This register reflects the CRC status of data line[7:0]. This register is only for MSDC Read. 1'b0: No CRC error 1'b1: CRC error

112c0070 EMMC_CFG0 EMMC Configuration Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BTSUP	BTWDLY											BTACHKDIS	BTMOD	BTSTOP	BTSTART
Type	RW	RW											RW	RW	WO	WO
Reset	0	0	0	0									0	0	0	0

Bit(s)	Mnemonic	Name	Description
15	**BTSUP**	BOOT_SUPPORT	**eMMC boot up support** The register bit indicates that boot mode is supported or not 1'b0: Not Support 1'b1: Support
14:12	**BTWDLY**	BOOT_WAIT_DELAY	**eMMC wait delay time** The register bit set the delay time to wait MMC device to exit boot up mode after boot stop bit is set. 3'b000: 0x1024 clock cycles 3'b001: 1x1024 clock cycles 3'b010: 2x1024 clock cycles 3'b111: 7x1024 clock cycles
3	**BTACHKDIS**	BOOT_ACK_CHK_DIS	**eMMC boot up mode ACK check Disable** 1'b0: Do ACK pattern check 1'b1: Bypass ACK pattern check
2	**BTMOD**	BOOT_MODE	**eMMC boot up mode** There are two kinds of boot up mode supported by eMMC 4.4. Reset CMD mode is option for eMMC 4.3. 1'b0: Pull low CMD mode 1'b1: Reset CMD mode
1	**BTSTOP**	BOOT_STOP	**eMMC boot up mode stop** The register bit is indicated that boot stop signal, read always return 0.
0	**BTSTART**	BOOT_START	**eMMC boot up start signal trigger** The register bit is boot up start signal trigger. read always return 0.

112c0074 EMMC_CFG1 EMMC Configuration Register 1 00200003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BTATOC												BTDTOC[19:16]			
Type	RW												RW			
Reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BTDTOC[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:20	**BTATOC**	BOOT_ACK_TOC	**eMMC ack pattern time out counter** in unit of 2^16 serial clock. SW could not set it to 12'hFFF.
19:0	**BTDTOC**	BOOT_DAT_TOC	**eMMC read boot data time out counter** in unit of 2^16 serial clock. SW could not set it to 20'hFFFFF.

112c0078 EMMC_STS EMMC Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										BTDRCV	BTARCV	BTSTS	BTATO	BTDTO	BTAERR	BTDERR
Type										RU	W1C	RU	W1C	W1C	W1C	W1C
Reset										0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
6	**BTDRCV**	BOOT_DAT_RECV	**eMMC boot data is received** This register is for S/W to check 1st 4-byte boot data is received or not. For other data after 1st 4B data, this bit should not be referenced. S/W should check RXFIFOCNT instead. 1'b0: There's no data in RXFIFO 1'b1: 1st 4B data is in RXFIFO
5	**BTARCV**	BOOT_ACK_RECV	**eMMC ack is received** Also need to check BOOT_ACK_ERR to determine pass or fail 1'b0: No ACK pattern is received 1'b1: ACK pattern has been received
4	**BTSTS**	BOOT_UP_STATE	**eMMC boot up mode status** The register bit indicates if MMC device operating in boot up mode state. 1'b0: Not in Boot up state 1'b1: Boot up state is on-going
3	**BTATO**	BOOT_ACK_TO	**eMMC ack timeout** The register bit indicates the controller detect a time out condition while waiting for an ack pattern on DAT0. 1'b0: No ACK pattern timeout error 1'b1: ACK pattern timeout error
2	**BTDTO**	BOOT_DAT_TO	**eMMC data timeout** The register bit indicates the controller detect a time out condition while waiting for boot data. 1'b0: No Data timeout error 1'b1: Data timeout error
1	**BTAERR**	BOOT_ACK_ERR	**eMMC ack error** The register bit indicates the status of ack pattern checking result. The bit is setting to 1 when ack pattern error. 1'b0: No ACK pattern check error 1'b1: ACK pattern check error
0	**BTDERR**	BOOT_CRC_ERR	**eMMC CRC error** The register bit indicates the CRC status of boot data. The bit is setting to 1 when data CRC error. 1'b0: No Data CRC error 1'b1: Data CRC error

112c007C EMMC_IOCON EMMC IO Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																

Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BTRST
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0	**BTRST**	BOOT_RST	**eMMC device boot up mode reset** The register bit is to trigger HW reset to set eMMC device entering into pre-idle state. 1'b0: de-assert RST_n to eMMC card 1'b1: Assert RST_n to eMMC card

112c0080 SD_ACMD_RESP SD ACMD Response Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMDRESP[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMDRESP[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMDRESP**	AUTOCMD_RESP	**SD Auto command response register** This register stores the response[39:8] of ACMD12/ACMD23/ACMD19.

112c0084 SD_ACMD19_TRG SD ACMD19 Target Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													ACMDFTSEL			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**ACMDFTSEL**	FINE_TUNE_SEL	**SD Auto command 19 test target selection** After auto-command 19 is triggered, MSDC will only change the phase of the selected one and keep the value of other tuning registers. S/W can get the result from AUTOCMD_STS. There are total 32 phases for each delay line. 4'd0: Select PAD_CLK_TXDLY[5:0] as the target to run auto-command19. 4'd1: Select PAD_CMD_RXDLY[5:0] as the target to run auto-

Bit(s)	Mnemonic	Name	Description
			command19. 4'd2: Select PAD_DAT_RD_RXDLY[5:0] as the target to run auto-command19. 4'd3: Select PAD_DAT_WR_RXDLY[5:0] as the target to run auto-command19. 4'd4: Select DAT0_RD_DLY[5:0] as the target to run auto-command19. 4'd5: Select DAT1_RD_DLY[5:0] as the target to run auto-command19. 4'd6: Select DAT2_RD_DLY[5:0] as the target to run auto-command19. 4'd7: Select DAT3_RD_DLY[5:0] as the target to run auto-command19. 4'd8: Select DAT4_RD_DLY[5:0] as the target to run auto-command19. 4'd9: Select DAT5_RD_DLY[5:0] as the target to run auto-command19. 4'd10: Select DAT6_RD_DLY[5:0] as the target to run auto-command19. 4'd11: Select DAT7_RD_DLY[5:0] as the target to run auto-command19. 4'd12: Select CMD_RESP_RXDLY[5:0] as the target to run auto-command19. Others: Reserved

112c0088 SD_ACMD19_STS SD ACMD19 Status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACMD19STS[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACMD19STS[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ACMD19STS**	AUTOCMD19_STS	**SD Auto command 19 test result register** When auto-command 19 is enabled, H/W will automatically try 32 times of command-19 and store the result into this register. This register contains 1st to 32th results in bit[0:31]

112c008C DMA_SA_HIGH4BIT DMA Current Address Resgiter of high 4bit 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													DMASAHIGH4BIT			
Type													RW			
Reset													0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0	**DMASAHIGH4BIT**	DMA_SURR_ADDR_HIGH4BIT	**it is usend to set high 4bit address of start address because 64G dram need 36bit address**

112c0090 DMA_SA DMA Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMASA[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMASA[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMASA**	DMA_STR_ADDR	**The start address of the DMA address** This register is used to set the start address of the DMA. In DMA basic mode, this field indicates the source or destination address of the data transfer which depends on the command. In descriptor base DMA, this is the descriptor chain start address.

112c0094 DMA_CA DMA Current Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DMACA[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMACA[15:0]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**DMACA**	DMA_CURR_ADDR	**The current address of the DMA address** This register is used to read the current address of the DMA descriptor chain.

112c0098 DMA_CTRL DMA Control Register 00006008

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		BSTSZ			SPLIT1K	LASTBF	DMAALIGN	DMAMOD					READYM	DMARSM	DMASTOP	DMASTART
Type		RW			RW	RW	RW	RW					RO	WO	A0	WO
Reset		1	1	0	0	0	0	0					1	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12	**BSTSZ**	BURST_SIZE	**DMA burst size**

Bit(s)	Mnemonic	Name	Description
			This field is used to specify the maximum transfer bytes allowed at the device per DMA burst. This field can not be modified when the DMA status is 1. 3'd3: 8 Bytes 3'd4: 16 Bytes 3'd5: 32 Bytes 3'd6: 64 Bytes Other: Reserved
11	**SPLIT1K**	DMA_SPLIT_1K	**This field is used to specify whether split burst when corss 1K boundry address** 1'b0: 1K boundary not split 1'b1: 1K boundary split
10	**LASTBF**	LAST_BUF	**Last buffer of the basic DMA mode** This field indicates the last buffer in the basic DMA mode
9	**DMAALIGN**	DMA_ALIGN	**This field is used to specify whether address aligment burst size** 1'b0: do not DAM burst size alignment 1'b1: DAM burst size alignment
8	**DMAMOD**	DMA_MODE	**DMA operation mode** This field indicates operation mode of DMA 1'b0: Basic DMA mode 1'b1: Descriptor base DMA mode
3	**READYM**	AHB_READYM	**only for debug when dma hang,sw can check if ahb bus is ok when gdma is hang** 1: bus is normal 0: bus not normal
2	**DMARSM**	DMA_RESUME	**DMA resume control register** This bit is used to resume the DMA transaction. Read always return 0
1	**DMASTOP**	DMA_STOP	**DMA Stop control register** This bit is used to stop the DMA transaction. When SW issue STOP command, SW must wait this bit de-assert or DMA inactive to guarantee stop done.
0	**DMASTART**	DMA_START	**DMA start control register** This bit is used to start the DMA transaction. Read always return 0

112c009C **DMA_CFG** **DMA Configuration Register** **00000004**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																DMACHKSUM12B
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MSDCACTIVEEN				AHBHPROT2EN							LOCKDISABLE	DSCPCSEN	DMASTS
Type			RW				RW							RW	RW	RU
Reset			0	0			0	0						1	0	0

Bit(s)	Mnemonic	Name	Description
16	**DMACHKSUM12B**	DMA_CHK_SUM_12B	**This register indicates GPD/BD checksum cover 16byte or 12byte**

Bit(s)	Mnemonic	Name	Description
			1'b0: GPD/BD checksum cover 16byte 1'b1: GPD/BD checksum only cover 12byte
13:12	**MSDCACTIVEEN**	MSDC_ACTIVE_EN	**This register will indicate how to control msdc_active** 2'b00: dynamic control msdc_active 2'b01: msdc_active = 0 2'b10: msdc_active = 1 2'b11: Reserved
9:8	**AHBHPROT2EN**	AHB_HPROT_2_EN	**This register will determine how to control hprot_2 pin of AHB bus** AHB_HPROT_2_EN = 2'b00, and Basic DMA Mode All the write transfers of a burst will access by bufferable mode except the last burst of DMA AHB_HPROT2_2_EN=2'b00, and Descriptor DMA Mode all the write transfers of a burst will access by bufferable mode except HW own update transfer 2'b00: dynamic control hprot_2 2'b01: hprot_2 = 0 2'b10: hprot_2 = 1
2	**LOCKDISABLE**	LOCK_DISABLE	**should disable lock in order to improve emi efficient** 1'b0: enable ahb lock 1'b1: disable ahb lock
1	**DSCPCSEN**	DMA_DSCP_CS_EN	**DMA descriptor checksum enable** This bit is used to enable or disable the descriptor checksum validation function for the descriptor. This field can not be modified when the DMA status is 1.
0	**DMASTS**	DMA_STATUS	**DMA status** This bit is used to indicate the status of the DMA. 1'b0: DMA engine is inactive 1'b1: DMA engine is active

112c00A0 SW_DBG_SEL MSDC S/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGSEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0	**SWDBGSEL**	DBG_SEL	**MSDC debug selection** This contain is reserved!

112c00A4 SW_DBG_OUT MSDC S/W Debug Output Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SWDBGO[31:16]															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SWDBGO[15:0]															

Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**SWDBGO**	DBG_OUT	**MSDC debug output** 32 bit output selected by SW_DBG_SEL register

112c00A8 DMA_LENGTH DMA Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	XFSZ[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	XFSZ[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**XFSZ**	XFER_SIZE	**DMA total transfer size** This field is used to specify the number of DMA transfer byte required for the movement of source data through DMA. This field is only valid in basic DMA mode.

112c00B0 PATCH_BIT0 MSDC Patch Bit Register 0 403C0006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PTCH 31	PTCH 30	PTCH 29	PTCH 28	PTCH 27	PTCH 26	PTCH22				PTCH18				PTCH 17	PTCH 16
Type	RW	RW	RW	RW	RW	RW	RW				RW				RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PTCH 15	CKGDLYS					INTCKS			DESC UP				PTCH 02	PTCH 01	
Type	RW	RW					RW			RW				RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0				1	1	

Bit(s)	Mnemonic	Name	Description
31	**PTCH31**	EN_MMC_DRV_RESP	**Enable MSDC always drives bus when output wakeup response (BREAK)** 1'b0: Disable 1'b1: Enable
30	**PTCH30**	DETECT_WR_CRC_TIMEOUT	**MSDC write data CRC phase timeout detection** 1'b0: Not detect CRC phase timeout 1'b1: detect CRC phase timeout
29	**PTCH29**	SPC_ALWAYS_PUSH	**SPC Buffer push mechanism** 1'b0: Push the buffer only when read transfer is on-going 1'b1: Always push the buffer
28	**PTCH28**	SDIO_INT_DLY_SEL	**SDIO interrupt latch time selection** 1'b0: Latch the data line value in internal SDIO interrupt period 1'b1: Latch the data line value in 1 clock delay of internal SDIO interrupt period

Bit(s)	Mnemonic	Name	Description
27	**PTCH27**	SDC_CMD_CMDFAIL_SEL	**SDIO interrupt period recovery selection** 1'b0: SDIO interrupt period will re-start after a CMD12 or CMD52 command is issued 1'b1: SDIO interrupt period whenever DAT line is not busy
26	**PTCH26**	SDC_CMD_IDRT_SEL	**SD identification response time selection** The register bit indicates if the command has a response with NID (that is, 5 serial clock cycles as defined in SD Memory Card Specification Part 1 Physical Layer Specification version 1.0) response time. The register bit is valid only when the command has a response token. Thus the register bit must be set to 1 for CMD2 (ALL_SEND_CID) and ACMD41 (SD_APP_OP_CMD). 1'b0: Otherwise. 1'b1: The command has a response with NID response time.
25:22	**PTCH22**	SDC_CFG_WDOD	**SD Write Data Output Delay** The period from finish of the response for the initial host write command or the last write data block in a multiple block write operation to the start bit of the next write data block requires at least two serial clock cycles. The register field is used to extend the period (Write Data Output Delay) in unit of one serial clock. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
21:18	**PTCH18**	SDC_CFG_BSYDLY	**SD R1B busy detection mode** The register field is only valid for the commands with R1b response. If the command has a response of R1b type, MS/SD controller must monitor the data line 0 for card busy status from the bit time that is two serial clock cycles after the command end bit to check if operations in SD/MMC Memory Card have finished. The register field is used to expand the time between the command end bit and end of detection period to detect card busy status. If time is up and there is no card busy status on data line 0, then the controller will abandon the detection. 4'b0000: No extend. 4'b0001: Extend one more serial clock cycle. 4'b0010: Extend two more serial clock cycles. 4'b1111: Extend fifteen more serial clock cycle.
17	**PTCH17**	SDIO_CFG_INTC_SEL	**SDIO Interrupt model selection** 1'b0: Only when data line [1] = 0 and then trigger SDIO interurpt event 1'b1: Only when data line [3:0] = 4'b1101 and then trigger SDIO interurpt event
16	**PTCH16**	MSDC_BLKNUM_SEL	**Configuration support ACMD23 reliable/force prog etc. feature** 1'b0: Support ACMD23 reliable/force prog etc. feature 1'b1: Don't support ACMD23 reliable/force prog etc. feature
15	**PTCH15**	MSDC_FIFO_RD_DIS	**MSDC RXFIFO Read Disable** 1'b0: Disable FIFO read permission to RXFIFO in PIO mode 1'b1: Enable FIFO read permission to RXFIFO in PIO mode
14:10	**CKGDLYS**	CKGEN_MSDC_DLY_SEL	**CKBUF in CKGEN Delay Selection** Total 32 stages
9:7	**INTCKS**	INT_DAT_LATCH_CK_SEL	**Internal MSDC clock phase selection** Total 8 stages, each stage can delay 1 clock period of msdc_src_ck
6	**DESCUP**	DESC_UP_SEL	**sd transfer done int should be issue when GPD have been update** 1'b1: enable new function for generate sd transfer done int 1'b0: use old function for generate sd transfer done int

Bit(s)	Mnemonic	Name	Description
2	**PTCH02**	DIS_REFLECT_CMDWR_WHEN_BSY	**Enable SD command register write montior** 1'b0: Enable monitor function 1'b1: Disable monitor function
1	**PTCH01**	EN_SDC_ODD_8BIT_SUP	**Enable SD odd number support for 8-bit data bus** 1'b0: Disable 1'b1: Enable

112c00B4 PATCH_BIT1 MSDC Patch Bit Register 1 FFFE0009

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MSHBFCKEN	MRCTLCKEN	MWCTLCKEN	MSDCKEN	MACMDCKEN	MVOLDTCKEN	MPSCCKEN	MSPCCKEN	HGDMACKEN	AXIWRAPCKEN	DCMEN	DCMDIVSEL1	DCMDIVSEL2		FORCESTOP	SINGLEBURST
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESETGDMA		BIAS28R0	BIAS28R1	BIAS28R2				GETCRCMARGIN	GETBUSYMARGIN	CMDTA			WRTA		
Type	RW		RW	RW	RW				RW	RW	RW			RW		
Reset	0		0	0	0	0	0	0	0	0	0	0	1	0	0	1

Bit(s)	Mnemonic	Name	Description
31	**MSHBFCKEN**	MSDC_CK_SHBFF_CKEN	**msdc_src_ck clock enable bit for SHBFF** 1'b0: Disable 1'b1: Enable
30	**MRCTLCKEN**	MSDC_CK_RCTL_CKEN	**msdc_src_ck clock enable bit for RCTL** 1'b0: Disable 1'b1: Enable
29	**MWCTLCKEN**	MSDC_CK_WCTL_CKEN	**msdc_src_ck clock enable bit for WCTL** 1'b0: Disable 1'b1: Enable
28	**MSDCKEN**	MSDC_CK_SD_CKEN	**msdc_src_ck clock enable bit for SD** 1'b0: Disable 1'b1: Enable
27	**MACMDCKEN**	MSDC_CK_ACMD_CKEN	**msdc_src_ck clock enable bit for ACMD** 1'b0: Disable 1'b1: Enable
26	**MVOLDTCKEN**	MSDC_CK_VOLDET_CKEN	**msdc_src_ck clock enable bit for VOLDET** 1'b0: Disable 1'b1: Enable
25	**MPSCCKEN**	MSDC_CK_PSC_CKEN	**msdc_src_ck clock enable bit for PSC** 1'b0: Disable 1'b1: Enable
24	**MSPCCKEN**	MSDC_CK_SPC_CKEN	**msdc_src_ck clock enable bit forSPC** 1'b0: Disable 1'b1: Enable
23	**HGDMACKEN**	AHB_CK_GDMA_CKEN	**hclk_ck clock enable bit for GDMA** 1'b0: Disable 1'b1: Enable
22	**AXIWRAPC**	AXI_WRAP_CKEN	**axi_ck and ahb_ck enbale bit for ahb2axi wrapper.must set this bit to 1 when in dma mode for emmc50 spec,or this bit is**

Bit(s)	Mnemonic	Name	Description
	KEN		**reserverd** 1'b0: disable 1'b1: enable
21	**DCMEN**	DCM_EN	**host enable dcm for low power,when there is no transfer the ahb_ck and msdc_src_ck will divder from spurce clk** 1'b1: disable 1'b0: enable
20	**DCMDIVSEL1**	DCM_DIV_SEL1	**for spec4.5 divider msdc_src_ck from source clk,for spec50 divider ahb_ck from source clk** 1'b1: 32 divder from source clk 1'b0: 16 divder from source clk
19:18	**DCMDIVSEL2**	DCM_DIV_SEL2	**this bit is reserved for spec45 only used for spec50** 2'b00: 64 divider for msdc_src_ck from source clk 2'b01: 32 divider for msdc_src_ck from source clk 2'b10: 16 divider for msdc_src_ck from source clk 2'b11: 128 divider for msdc_src_ck from source clk
17	**FORCESTOP**	FORCE_STOP_GDMA	**this bit can force state from WDMI_LAST_DATA to WDMI_IDLE with stop dma bit** 1'b1: enable stop wdmi_cs when state hang in WDMI_LAST_DATA with stop dma bit 1'b0: disable
16	**SINGLEBURST**	ENABLE_SINGLE_BURST	**the ahb bus will not support incr1 burst type in future. And it will only affect AHB bus msdc design,not affect AXI bus design** 1'b0: hw will send incr1 burst type 1'b1: hw will send single burst typr instead of incr1 type
15	**RESETGDMA**	RESET_GDMA	**sw can sw reset gdma when design hang** 1 'b1: reset gdma 1'b0: not reset gdma
13	**BIAS28R0**	BIAS_EXTBIAS_28NM	**28NM BIAS Controler register 0**
12	**BIAS28R1**	BIAS_EN18IO_28NM	**28NM BIAS Controler register 1**
11:8	**BIAS28R2**	BIAS_TUNE_28NM	**28NM BIAS Controler register 2**
7	**GETCRCMARGIN**	GET_CRC_MARGIN	**it will add margin for get crc status when card resp crc not match spec 2cycle from endbit** 1'b0: 8 cycle reserved for get crc status from write data crc endbit 1'b1: 16 cycle reserved for get crc status from write data crc endbit
6	**GETBUSYMARGIN**	GET_BUSY_MARGIN	**it will add margin for get busy state of data0** 1'b0: 1 cycle reserved for get busy state from src status endbit 1'b1: 3cycle reserved for get busy state from src status endbit
5:3	**CMDTA**	CMD_RSP_TA_CNTR	**CMD response turn around period** The turn around cycle = CMD_RSP_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode, this register should be set to 0
2:0	**WRTA**	WRDAT_CRCS_TA_CNTR	**Write data and CRC status turn around period** The turn around cycle = WRDAT_CRCS_TA_CNTR + 2 In USH104 mode, this register should be set to 1 In non-UHS104 mode,this register should be set to 0

112c00B8 PATCH_BIT2 MSDC Patch Bit Register 2 14801803

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	CRCSTSENSEL			CFGCRCSTS	CFGCRCSTSCNT		CFGCRCSTSEDGE	CFG_CRCSTS_SEL	POPENCNT					RESPSTSENSEL		
Type	RW			RW	RW		RW	RW	RW					RW		
Reset	0	0	0	1	0	1	0	0	1	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFGRESP	CFGRESPCNT			INTCRESPSEL		CFGRDAT	CFGRDATCNT					RESPWAITCNT		SUPPORT64G	ENHANCEGPD
Type	RW	RW			RW		RW	RW					RW		RW	RW
Reset	0	0	0	1	1		0	0	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
31:29	**CRCSTSENSEL**	CRCSTS_LATCH_EN_SEL	**This register used configuration latch CRC Status enable signal for async fifo in emmc45** 3'b000: latch CRC Status enable signal not delay 3'b001: latch CRC Status enable signal delay 1T msdc_ck 3'b010: latch CRC Status enable signal delay 2T msdc_ck 3'b011: latch CRC Status enable signal delay 3T msdc_ck 3'b111: latch CRC Status enable signal delay 7T msdc_ck
28	**CFGCRCSTS**	CFG_CRCSTS	**This register used configuration CRC Status path selection,this setting only used emmc4.5 feature** 1'b0: Latch CRC Status select delay-line path 1'b1: Latch CRC Status select async fifo path
27:26	**CFGCRCSTSCNT**	CFG_CRCSTS_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 2'b00: push 0 data in async fifo when start pop out data from async fifo 2'b01: push 1 data in async fifo when start pop out data from async fifo 2'b10: push 2 data in async fifo when start pop out data from async fifo 2'b11: push 3 data in async fifo when start pop out data from async fifo
25	**CFGCRCSTSEDGE**	CFG_CRCSTS_EDGE	**This register configuration used rising async fifo or falling async fifo** 1'b0: async fifo latch CRC Status used rising async fifo 1'b1: async fifo latch CRC Status used falling async fifo
24		CFG_CRCSTS_SEL	**This register configuration async fifo path selection** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
23:20	**POPENCNT**	POP_EN_CNT	**pop enable counter** This field is used to define how many write pointer and the read pointer margin began to pop data transfer
18:16	**RESPSTSENSEL**	RESP_LATCH_EN_SEL	**This register used configuration latch CMD Response enable signal for async fifo in emmc45** 3'b000: latch CMD Response enable signal not delay 3'b001: latch CMD Response enable signal delay 1T msdc_ck 3'b010: latch CMD Response enable signal delay 2T msdc_ck 3'b011: latch CMD Response enable signal delay 3T msdc_ck 3'b111: latch CMD Response enable signal delay 7T msdc_ck
15	**CFGRESP**	CFG_RESP	**This register used configuration CMD Response path selection,this setting only used emmc4.5 feature** 1'b0: Latch CMD Response select async fifo path 1'b1: Latch CMD Response select delay-line path

Bit(s)	Mnemonic	Name	Description
14:12	**CFGRESPCNT**	CFG_RESP_CNT	**This register used configuration how many data push in async fifo until start pop out data from async fifo ,this register setting min is 1 do not setting is 0,this setting only used emmc4.5 feature** 3'b000: push 0 data in async fifo when start pop out data from async fifo 3'b001: push 1 data in async fifo when start pop out data from async fifo 3'b111: push 7 data in async fifo when start pop out data from async fifo
11	**INTCRESPSEL**	INTC_RESP_SEL	**This register configuration BREAK command async fifo path** 1'b0: used normal path in async fifo 1'b1: used 2DFF path in async fifo
9	**CFGRDAT**	CFG_RDAT	**This register used configuration read data path** 1'b0: read data path by pass delay line 1'b1: read data path through delay line
8:4	**CFGRDATCNT**	CFG_RDAT_CNT	**This register used configuration read data path delay line**
3:2	**RESPWAITCNT**	RESP_WAIT_CNT	**This register used configuration cmd response timeout,timeout cycle is 65T+16*RESP_WAIT_CNT** 2'b00: cmd response timeout is 65T 2'b01: cmd response timeout is 65T+ 16*1T 2'b01: cmd response timeout is 65T+ 16*2T 2'b01: cmd response timeout is 65T+ 16*3T
1	**SUPPORT64G**	SUPPORT_64G	**This register used which proj support high 64G dram space access** 1'b1: support 64G dram access 1'b0: not support 64G dram access
0	**ENHANCEGPD**	ENHANCE_WAIT_GPD	**if sw clear int when gpd not update finish,design will hang.so you can set this bt to 1 to avoid this iuuse in enhance write mode** 1'b1: use new HW code for update gpd in enhance mode 1'b0: use old HW code

112c00F0 **PAD_TUNE0** **MSDC Pad Tuning Register0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CLKTDLY					CMDRRDLY					CMDRRDLYSEL	CMDRDLY				
Type	RW					RW					RW	RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDLYSEL		DATRRDLYSEL	DATRRDLY					DELAYEN			DATWRDLY				
Type	RW		RW	RW					RW			RW				
Reset	0		0	0	0	0	0	0	0			0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:27	**CLKTDLY**	PAD_CLK_TXDLY	**CLK Pad TX Delay Control** This register is used to add delay to CLK phase. Total 32 stages
26:22	**CMDRRDLY**	PAD_CMD_RESP_	**CMD Response Internal Delay Line Control**

Bit(s)	Mnemonic	Name	Description
		RXDLY	This register is used to fine-tune response phase latched by MSDC internal clock Total 32 stages
21	**CMDRRDLYSEL**	PAD_CMD_RD_RXDLY_SEL	**Decide CMD Response pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY**	PAD_CMD_RXDLY	**CMD Pad RX Delay Line1 Control** This register is used to fine-tune CMD pad macro respose latch timing Total 32 stages
15	**RXDLYSEL**	PAD_RXDLY_SEL	**Decide rx delay line tune data path or clock path** 1'b0: rx delay line tune data path 1'b1: rx delay line tune clock path
13	**DATRRDLYSEL**	PAD_DAT_RD_RXDLY_SEL	**Decide rx data pass through data delay line1 or not** 1'b0: pass 1'b1: do not pass
12:8	**DATRRDLY**	PAD_DAT_RD_RXDLY	**DAT Pad RX Delay Line1 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages
7	**DELAYEN**	DELAY_EN	**enable all delay cell toogle when power on** 1'b0: disable delay cell toogle default 1'b1: enable delay cell toogle default
4:0	**DATWRDLY**	PAD_DAT_WR_RXDLY	**Write Data Status Internal Delay Line Control** This register is used to fine-tune write status phase latched by MSDC internal clock Total 32 stages

112c00F4 PAD_TUNE1 MSDC Pad Tuning Register1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											CMDRRDLY2SEL	CMDRDLY2				
Type											RW	RW				
Reset											0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DATRRDLY2SEL	DATRRDLY2												
Type			RW	RW												
Reset			0	0	0	0	0	0								

Bit(s)	Mnemonic	Name	Description
21	**CMDRRDLY2SEL**	PAD_CMD_RD_RXDLY2_SEL	**Decide CMD Response pass through data delay line2 or not** 1'b0: pass 1'b1: do not pass
20:16	**CMDRDLY2**	PAD_CMD_RXDLY2	**CMD Pad RX Delay Line2 Control** This register is used to fine-tune CMD pad macro respose latch timing in data path Total 32 stages
13	**DATRRDLY2SEL**	PAD_DAT_RD_RXDLY2_SEL	**Decide rx data pass through data delay line2 or not** 1'b0: pass

Bit(s)	Mnemonic	Name	Description
			1'b1: do not pass
12:8	**DATRRDLY2**	PAD_DAT_RD_RXDLY2	**DAT Pad RX Delay Line2 Control (for MSDC read only)** This register is used to fine-tune DAT pad macro read data latch timing Total 32 stages

112c00F8 DAT_RD_DLY0 MSDC Data Delay Line Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY								DAT1RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY								DAT3RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY**	DAT0_RD_DLY	**DAT0 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY**	DAT1_RD_DLY	**DAT1 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
12:8	**DAT2RDDLY**	DAT2_RD_DLY	**DAT2 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY**	DAT3_RD_DLY	**DAT3 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

112c00FC DAT_RD_DLY1 MSDC Data Delay Line Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY								DAT5RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY								DAT7RDDLY				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY**	DAT4_RD_DLY	**DAT4 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY**	DAT5_RD_DLY	**DAT5 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY**	DAT6_RD_DLY	**DAT6 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY**	DAT7_RD_DLY	**DAT7 Pad RX Delay Line1 Control (for MSDC RD)** Total 32 stages

112c0100 DAT_RD_DLY2 MSDC Data Delay Line Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT0RDDLY2								DAT1RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT2RDDLY2								DAT3RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT0RDDLY2**	DAT0_RD_DLY2	**DAT0 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT1RDDLY2**	DAT1_RD_DLY2	**DAT1 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT2RDDLY2**	DAT2_RD_DLY2	**DAT2 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT3RDDLY2**	DAT3_RD_DLY2	**DAT3 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

112c0104 DAT_RD_DLY3 MSDC Data Delay Line Register 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DAT4RDDLY2								DAT5RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				DAT6RDDLY2								DAT7RDDLY2				
Type				RW								RW				
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
28:24	**DAT4RDDLY2**	DAT4_RD_DLY2	**DAT4 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
20:16	**DAT5RDDLY2**	DAT5_RD_DLY2	**DAT5 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
12:8	**DAT6RDDLY2**	DAT6_RD_DLY2	**DAT6 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages
4:0	**DAT7RDDLY2**	DAT7_RD_DLY2	**DAT7 Pad RX Delay Line2 Control (for MSDC RD)** Total 32 stages

112c0110 HW_DBG_SEL MSDC H/W Debug Selection Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		DBGWSEL	DBG3SEL						DBG2SEL							
Type		RW	RW						RW							
Reset		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DBG1SEL						DBG0SEL							
Type			RW						RW							

Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30	**DBGWSEL**	HW_DBG_WRAP_SEL	**H/W debug output selection for wrapper** 0: Select original debug pins 1: Select wrapper debug pins
29:24	**DBG3SEL**	HW_DBG3_SEL	**H/W debug output selection**
23:16	**DBG2SEL**	HW_DBG2_SEL	**H/W debug output selection**
13:8	**DBG1SEL**	HW_DBG1_SEL	**H/W debug output selection**
7:0	**DBG0SEL**	HW_DBG0_SEL	**H/W debug output selection**

112c0114 MAIN_VER MSDC Main Version Register 20140512

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAINVER[31:16]															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAINVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	1	0	1	0	0	0	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:0	**MAINVER**	MAIN_VER	**Main Version**

112c0118 ECO_VER MSDC ECO Version Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECOVER[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECOVER[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0	**ECOVER**	ECO_VER	**ECO Version**

112c0180 EMMC50_PAD_CTL0 MSDC eMMC5.0 CLK PAD Control Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							DLN1		DLP1		DLN0		DLP0		HLSEL	DCCSEL
Type							RW		RW		RW		RW		RW	RW
Reset							0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
9:8	**DLN1**	DLN1	**The gear to tune the low duty cycle for first Duty Cycle Corrector,four gears from 00 to 11**
7:6	**DLP1**	DLP1	**The gear to tune the high duty cycle for second Duty Cycle Corrector,four gears from 00 to 11**
5:4	**DLN0**	DLN0	**The gear to tune the low duty cycle for first Duty Cycle Corrector,four gears from 00 to 11**
3:2	**DLP0**	DLP0	**The gear to tune the high duty cycle for first Duty Cycle Corrector,four gears from 00 to 11**
1	**HLSEL**	HL_SEL	**Decide to tune the low/high duty cycle** 1'b0: Tune high duty cycle 1'b1: Tune low duty cycle
0	**DCCSEL**	DCC_SEL	**Decide SCLK pass through duty cycle corrector or not** 1'b0: do not pass 1'b1: pass

112c0184 **EMMC50_PAD_DS_CTL0** **MSDC eMMC5.0 DS PAD Control Register 0** **0000001C**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														DSDRV		
Type														RW		
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DSTDSEL				DSRDSEL						DSSMT	DSIES	DSPUPD	DSR1	DSR0	DSSR
Type	RW				RW						RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0

Bit(s)	Mnemonic	Name	Description
18:16	**DSDRV**	PAD_DS_DRV	**DS Pad Driving Strength Control**
15:12	**DSTDSEL**	PAD_DS_TDSEL	**DS Pad TX duty select**
11:6	**DSRDSEL**	PAD_DS_RDSEL	**DS Pad RX duty select**
5	**DSSMT**	PAD_DS_SMT	**DS Pad RX input buffer schmit trigger hysteresis control enable.** High asserted. SMT=1, Schmit Trigger enable
4	**DSIES**	PAD_DS_IES	**DS Pad RX input buffer enable** High asserted. Data path from IO to O. 1'b0: Disable input buffer 1'b1: Enable input buffer
3	**DSPUPD**	PAD_DS_PUPD	**DS Pad pull-up(0)/pull-down(1) control**
2	**DSR1**	PAD_DS_R1	**DS Pad 50K resistor control**
1	**DSR0**	PAD_DS_R0	**DS Pad 10K resistor control**
0	**DSSR**	PAD_DS_SR	**DS Pad Output Slew Rate Control.** High asserted. 1'b0: no slew rate control 1'b1: slower slew

112c0188 **EMMC50_PAD_DS_TUNE** **MSDC eMMC5.0 DS Pad Tuning Register** **00014015**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																PAD_DS_DLY3[4:4]
Type																RW
Reset																1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PAD_DS_DLY3[3:0]				PAD_DS_DLY2					DSDLY1					DSDLY2SEL	DSDLYSEL
Type	RW				RW					RW					RW	RW
Reset	0	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
16:12		PAD_DS_DLY3	**The gear of the second delay line for DS for input data latch in data pad macro,32 gears from 0 to 31**
11:7		PAD_DS_DLY2	**The gear of the second delay line for DS for input data latch in data pad macro,32 gears from 0 to 31**
6:2	**DSDLY1**	PAD_DS_DLY1	**The gear of the first delay line for DS for input data latch in data pad macro,32 gears from 0 to 31**
1	**DSDLY2SEL**	PAD_DS_DLY2_SEL	**Select if DS pass through second delay macro** 1'b0: Do not pass 1'b1: Pass
0	**DSDLYSEL**	PAD_DS_DLY_SEL	**Select if DS pass through delay macro** 1'b0: Do not pass 1'b1: Pass

112c018C **EMMC50_PAD_CMD_TUNE** **MSDC eMMC5.0 CMD Pad Tuning Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						CMDTXDLY					CMDRXDLY3					CMDRXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
10:6	**CMDTXDLY**	PAD_CMD_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
5:1	**CMDRXDLY3**	PAD_CMD_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
0	**CMDRXDLY3MUXSEL**	PAD_CMD_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass

112c0190 EMMC50_PAD_DAT01_TUNE MSDC eMMC5.0 DAT0/1 Pad Tuning Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						DAT1TXDLY					DAT1RXDLY3					DAT1RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						DAT0TXDLY					DAT0RXDLY3					DAT0RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
26:22	**DAT1TXDLY**	PAD_DAT1_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
21:17	**DAT1RXDLY3**	PAD_DAT1_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
16	**DAT1RXDLY3MUXSEL**	PAD_DAT1_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass
10:6	**DAT0TXDLY**	PAD_DAT0_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
5:1	**DAT0RXDLY3**	PAD_DAT0_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
0	**DAT0RXDLY3MUXSEL**	PAD_DAT0_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass

112c0194 EMMC50_PAD_DAT23_TUNE MSDC eMMC5.0 DAT2/3 Pad Tuning Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						DAT3TXDLY					DAT3RXDLY3					DAT3RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						DAT2TXDLY					DAT2RXDLY3					DAT2RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
26:22	**DAT3TXDLY**	PAD_DAT3_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
21:17	**DAT3RXDLY**	PAD_DAT3_RX_DL	**The gear of delay line for RX data,32 gears from 0 to 31**

Bit(s)	Mnemonic	Name	Description
	3	Y3	
16	**DAT3RXDLY3MUXSEL**	PAD_DAT3_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass
10:6	**DAT2TXDLY**	PAD_DAT2_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
5:1	**DAT2RXDLY3**	PAD_DAT2_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
0	**DAT2RXDLY3MUXSEL**	PAD_DAT2_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass

112c0198 EMMC50_PAD_DAT45_TUNE MSDC eMMC5.0 DAT4/5 Pad Tuning Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						DAT5TXDLY					DAT5RXDLY3					DAT5RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						DAT4TXDLY					DAT4RXDLY3					DAT4RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
26:22	**DAT5TXDLY**	PAD_DAT5_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
21:17	**DAT5RXDLY3**	PAD_DAT5_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
16	**DAT5RXDLY3MUXSEL**	PAD_DAT5_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass
10:6	**DAT4TXDLY**	PAD_DAT4_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
5:1	**DAT4RXDLY3**	PAD_DAT4_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
0	**DAT4RXDLY3MUXSEL**	PAD_DAT4_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass

112c019C EMMC50_PAD_DAT67_TUNE MSDC eMMC5.0 DAT6/7 Pad Tuning Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						DAT7TXDLY					DAT7RXDLY3					DAT7RXDLY3MUXSEL

																Y3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						DAT6TXDLY					DAT6RXDLY3					DAT6RXDLY3MUXSEL
Type						RW					RW					RW
Reset						0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
26:22	**DAT7TXDLY**	PAD_DAT7_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
21:17	**DAT7RXDLY3**	PAD_DAT7_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
16	**DAT7RXDLY3MUXSEL**	PAD_DAT7_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass
10:6	**DAT6TXDLY**	PAD_DAT6_TX_DLY	**The gear of delay line for TX data,32 gears from 0 to 31**
5:1	**DAT6RXDLY3**	PAD_DAT6_RX_DLY3	**The gear of delay line for RX data,32 gears from 0 to 31**
0	**DAT6RXDLY3MUXSEL**	PAD_DAT6_RX_DLY3_MUX_SEL	**Select if the input data pass through the delay line** 1'b0: Do not pass 1'b1: Pass

112c0204 EMMC51_CFG0 eMMC51 configuration register0 000350D4

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											CMDQCMDEN	RDATCNT[9:5]				
Type											RW	RW				
Reset											0	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RDATCNT[4:0]					WDATCNT										CMDQEN
Type	RW					RW										RW
Reset	0	1	0	1	0	0	0	0	1	1	0	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description
21	**CMDQCMDEN**	CMDQ_CMD_ENABLE	**configuration cmd queue enable type** 1'b0: cmd queue enable for cmd13,cmd44,cmd45 1'b1: cmd queue enable for all cmd
20:11	**RDATCNT**	RDAT_CNT	**read data counter threshold for cmd queue**
10:1	**WDATCNT**	WDAT_CNT	**write data counter threshold for cmd queue**
0	**CMDQEN**	CMDQ_EN	**cmd queue enable**

112c0208 EMMC50_CFG0 eMMC50 configuration register0 02E889E2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name			TXSKEWSEL	RDVALIDSEL	WRVALIDSEL	MSDCRDVALID	MSDCWRVALID	EMMC50MONSEL	READDATCNT			ENDBITCNT[9:5]				
Type			RW	RW	RW	RU	RU	RW	RW			RW				
Reset			0	0	0	0	1	0	1	1	1	0	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENDBITCNT[4:0]					CMDEDGESEL	CMDRESPSEL	END_BITCHKCNT				CRCSTSSEL	CRCSTSEDGE	CRCSTSCNT		PADLATCK
Type	RW					RW	RW	RW				RW	RW	RW		RW
Reset	1	0	0	0	1	0	0	1	1	1	1	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
29	**TXSKEWSEL**	MSDC_TXSKEW_SEL	1'b0: clk/cmd/data output not through DFF in hs400 mode 1'b1: clk/cmd/data output through DFF in hs400 mode
28	**RDVALIDSEL**	MSDC_RD_VALID_SEL	
27	**WRVALIDSEL**	MSDC_WR_VALID_SEL	
26	**MSDCRDVALID**	MSDC_RD_VALID	**for emmc50 spec,host cant stop pad clk in one block transfer,so host must have whole block len data in fifo when host want transfer data to device** 1'b0: the sd fifo is not valid to read for writing device 1'b1: the sd fifo is valid to read for writing device
25	**MSDCWRVALID**	MSDC_WR_VALID	**for emmc50 spec,host cant stop pad clk in one block transfer,so host must have enough space for storing a block len data read from device** 0: the sd fifo is not valid for reading data from device 1: the sd fifo is valid for reading data from device
24	**EMMC50MONSEL**	EMMC50_MON_SEL	**we can debug emmc50 signals through set this bit 1 when debug** 0: chose emmc50 debug signals to debug pin 1: not chose emmc50 debug signals to debug pins
23:21	**READDATCNT**	READ_DAT_CNT	**setting latch how many data in RCLK domain switch pop condition**
20:11	**ENDBITCNT**	ENDBIT_CNT	**setting endbit check value in RCLK domain**
10	**CMDEDGESEL**	CMD_EDGE_SEL	**CMD Response edge selsection** 1'b0: CMD Response from RCLK rising edge latch 1'b1: CMD Response from RCLK falling edge latch
9	**CMDRESPSEL**	CMD_RESP_SEL	**CMD Response output selsection** 1'b0: CMD Response from delay line,this is the same as emmc4.5 spec design 1'b1: CMD Response from FIFO,this is design for emmc5.0 spec
8:5	**END_BITCHKCNT**	END_BIT_CHK_CNT	**setting how many cycles to check device busy status after CRC Status**
4	**CRCSTSSEL**	CRC_STS_SEL	**CRC Status output selection** 1'b0: CRC Status data from 2-DFF,before eMMC5.0 used 1'b1: CRC Status data from FIFO.eMMC5.0 used
3	**CRCSTSEDGE**	CRC_STS_EDGE	**in hs400 mode,select CRC Status** dat psh 1'b0: in hs400 mode select rising edge CRC Status data psh 1'b1: in hs400 mode select falling edge CRC Status data psh
2:1	**CRCSTSCNT**	CRC_STS_CNT	
0	**PADLATCK**	PAD_CMD_LATCH	**PAD CMD latch clock selection ,in HS400 mode**

Bit(s)	Mnemonic	Name	Description
		_CK	1'b0: select inyternal clock as pad cmd latch clock in HS400 mode 1'b1: select DS(RCLK) as pad cmd latch clock in HS400 mode

112c020C EMMC50_CFG1 eMMC50 configuration register1 0100070C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	reserve0			DSCFG	PSHPSSEL	PSHCNT			EMMC50DEBUGSEL							
Type	RW			RW	RW	RW			RW							
Reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WAIT8CLKCNT				RDATSTOP	CKSWITCHCNT			WRPTRMARGIN							
Type	RW				RW	RW			RW							
Reset	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	0

Bit(s)	Mnemonic	Name	Description
31:29		reserve0	**reserve for backup2**
28	**DSCFG**	DS_CFG	**This field is used to define wether mask redundant DS clock** 1'b0: mask redundant DS clock 1'b1: not mask redundant DS clock
27	**PSHPSSEL**	PSH_PS_SEL	**This field is used to define used rising or falling edge push pulse** 1'b0: select rising edge pulse 1'b1: select falling edge pulse
26:24	**PSHCNT**	PSH_CNT	**This field is used to define how many push cnt start sync pop pulse to pop data**
23:16	**EMMC50DEBUGSEL**	EMMC50_DEBUG_SEL	**emmc50 debug signal out selection**
15:12	**WAIT8CLKCNT**	WAIT_8_CLK_CNT	**configuration 8 clock counter**
11	**RDATSTOP**	RD_DAT_STOP	**eMMC50 read data stop signals slection** 1'b0: select rising signal 1'b1: select falling signal
10:8	**CKSWITCHCNT**	LATCH_CK_SWITCH_CNT	**after CRC Status,setting latchclock switch counter**
7:0	**WRPTRMARGIN**	WR_PTR_MARGIN	**write pointer marign**

112c021C EMMC50_CFG2 ahb2axi wrapper control register 0F0C0000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		AXIBUSY	RESPERRTYPE		SETLEN				RDOUTSTANDINGNUM					BOUND4K	BOUND2K	BOUND1K
Type		RO	RO		RW				RW					RW	RW	RW
Reset		0	0	0	1	1	1	1	0	0	0	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BOUND512B	BOUND256B	BOUND128B					SHAREENRDEMI	IOMMURDEMI	reserved1				SHAREENWREMI	IOMMUWREMI	
Type	RW	RW	RW					RW	RW	RW				RW	RW	
Reset	0	0	0					0	0	0	0	0	0	0	0	

Bit(s)	Mnemonic	Name	Description
30	**AXIBUSY**	AXI_BUSY	**check if axi bus is busy or idle** 1'b1: axi wrapper is busy 1'b0: axi wrapper is idle
29:28	**RESPERRTYPE**	AXI_RESP_ERR_TYPE	**when sw get axi respnse error int,you can check this bits to know which type error occr** 2'b00: no error 2'b01: do not care 2'b10: slave error 2'b11: decode error
27:24	**SETLEN**	AXI_SET_LEN	**sw can program this bits to set the transfer num in one axi burst both for read and write.for perfomance,this bits can set up to f** 4'b0000: 1 beats per one burst 4'b0001: 2 beats per one burst 4'b0010: 3 beats per one burst... 4'b1111: 16 beats per one burst
23:19	**RDOUTSTANDINGNUM**	AXI_RD_OUTSTANDING_NUM	**axi support outstanding transfer,it means when there is no reponse you can also sent axi cmd.for perfomance ,sw cant set this bits up to 13** 5'b: 00001:support 1 outstanding 5'b: 00002:support 2 outstanding 5'b: 00003:support 3 outstanding... 5'b: 01011:support 13 outstanding
18	**BOUND4K**	AXI_BOUND_4K	**default is 4K in axi spec** 1'b1: chose 4 K boundary 1'b0: not chose 4K boundary
17	**BOUND2K**	AXI_BOUND_2K	**for future project option** 1'b1: chose 2 K boundary 1'b0: not chose 2K boundary
16	**BOUND1K**	AXI_BOUND_1K	**for future project option** 1'b1: chose 42K boundary 1'b0: not chose 1K boundary
15	**BOUND512B**	AXI_BOUND_512B	**for future project option** 1'b1: chose 512B boundary 1'b0: not chose 512B boundary
14	**BOUND256B**	AXI_BOUND_256B	**for future project option** 1'b1: chose 256B boundary 1'b0: not chose 256B boundary
13	**BOUND128B**	AXI_BOUND_128B	**for future project option** 1'b1: chose 128B boundary 1'b0: not chose 128B boundary
8	**SHAREENRDEMI**	AXI_SHARE_EN_RD_EMI	1'b1: share enable 1'b0: not share enable
7	**IOMMURDEMI**	AXI_IOMMU_RD_EMI	1'b1: through iommu 1'b0: not through iommu
2	**SHAREENWREMI**	AXI_SHARE_EN_WR_EMI	1'b1: share enable 1'b0: not share enable
1	**IOMMUWREMI**	AXI_IOMMU_WR_EMI	1'b1: through iommu 1'b0: not through iommu

112c0220 EMMC50_CFG3 0F28A3C1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	reserved2			PREULTRASETRD						ULTRASETRD						PREULTRASETWR[5:5]
Type	RO			RW						RW						RW
Reset	0	0	0	0	1	1	1	1	0	0	1	0	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PREULTRASETWR[4:0]					ULTRASETWR						OUTSTANDINGWR				
Type	RW					RW						RW				
Reset	1	0	1	0	0	0	1	1	1	1	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
28:23	**PREULTRASETRD**	PREULTRA_SET_RD	**msdc gdma have more high priority when these bit set bigger**
22:17	**ULTRASETRD**	ULTRA_SET_RD	**msdc gdma have more high priority when these bit set bigger**
16:11	**PREULTRASETWR**	PREULTRA_SET_WR	**msdc gdma have more high priority when these bit set smaller**
10:5	**ULTRASETWR**	ULTRA_SET_WR	**msdc gdma have more high priority when these bit set smaller**
4:0	**OUTSTANDINGWR**	OUT_STANDING_WR	**axi support outstanding transfer,it means when there is no reponse you can also sent axi cmd.for improve perfomance ,sw cant set this bits up to 13**

112c0224 EMMC50_CFG4 **0001281E**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	reserved3									WRAP_SEL					ULTRAEN	
Type	RO									RW					RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IMPRULTRASETRD								IMPRULTRASETWR							
Type	RW								RW							
Reset	0	0	1	0	1	0	0	0	0	0	0	1	1	1	1	0

Bit(s)	Mnemonic	Name	Description
22:18	**WRAP_SEL**	wrap_sel	**for hw sel debug signal**
17:16	**ULTRAEN**	ULTRA_EN	**normally ,can not set to high priority,because it will reduce oher module performace** 2'b00: lowest priority 2'b01: lower priority 2'b10: middle priority 2'b11: high priority
15:8	**IMPRULTRASETRD**	IMPR_ULTRA_SET_RD	**msdc gdma have more high priority when these bit set bigger.host can improve priority at base of set ULTRA_SET_RD**
7:0	**IMPRULTRASETWR**	IMPR_ULTRA_SET_WR	**msdc gdma have more high priority when these bit set smaller.host can improve priority at base of set ULTRA_SET_WR**

112c0228 EMMC50_BLOCK_LENGTH **00000080**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								BLOCKLENGTH								
Type								RW								
Reset								0	1	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
8:0	**BLOCKLENGTH**	EMMC50_BLOCK_LENGTH	**sw set the fifo water level for stop clk in diff block length.** If device block length is 512B,this register should set 9'd128 If device block length is 256B,this register should set 9'd64 If device block length is 128B,this register should set 9'd32 If device block length is 64B,this register should set 9'd16 If device block length is 32B,this register should set 9'd8 this register is used for SDIO30+,other spec should use default value

14 NAND Flash Interface

14.1 Introduction

The NAND Flash Interface (NFI) and ECC engine (in NFI mode) can automatically generate ECC syndrome bits when programming or reading the device. If the user approves the way it stores the syndrome bits in the spare area for each page, the HW_ECC mode can be used. Otherwise, the user can prepare the data (may contains operating system information or ECC syndrome bits) for the spare area with another arrangement. In former cases, the NFI and ECC engine (in NFI mode) checks the syndrome bits when reading from the device. The ECC module features BCH code, which is capable of correcting up to 60-bits errors within one sector.

14.2 Feature list

NAND flash interface supports SLC/MLC NAND. The NAND FLASH interface supports the features below:

- Legacy NAND interface
- Toggle NAND(ddr1.0) interface
- ONFI NAND (2.x)interface
- ECC (BCH code) acceleration capable of 60-bit error correction (with ECC engine)
- Programmable page size(512B/2K/4K/8K/16K) ,sec_size(512B/1024B)and spare size(16B→128B)
- Word/byte access through APB bus
- DMA for massive data transfer
- Latch sensitive interrupt to indicate ready state for read, program and erase operation
- Programmable wait states, command/address setup and hold time, read enable hold time and write enable recovery time
- Supports 2-chip selection for NAND flash parts.
- Supports 8-bits Legacy I/O interface, 8bits TOGGLE/ONFI NAND I/O interface
- CRC16
- Randomizer(TOSHIBA/SAMSUNG)

14.3 Block Diagram

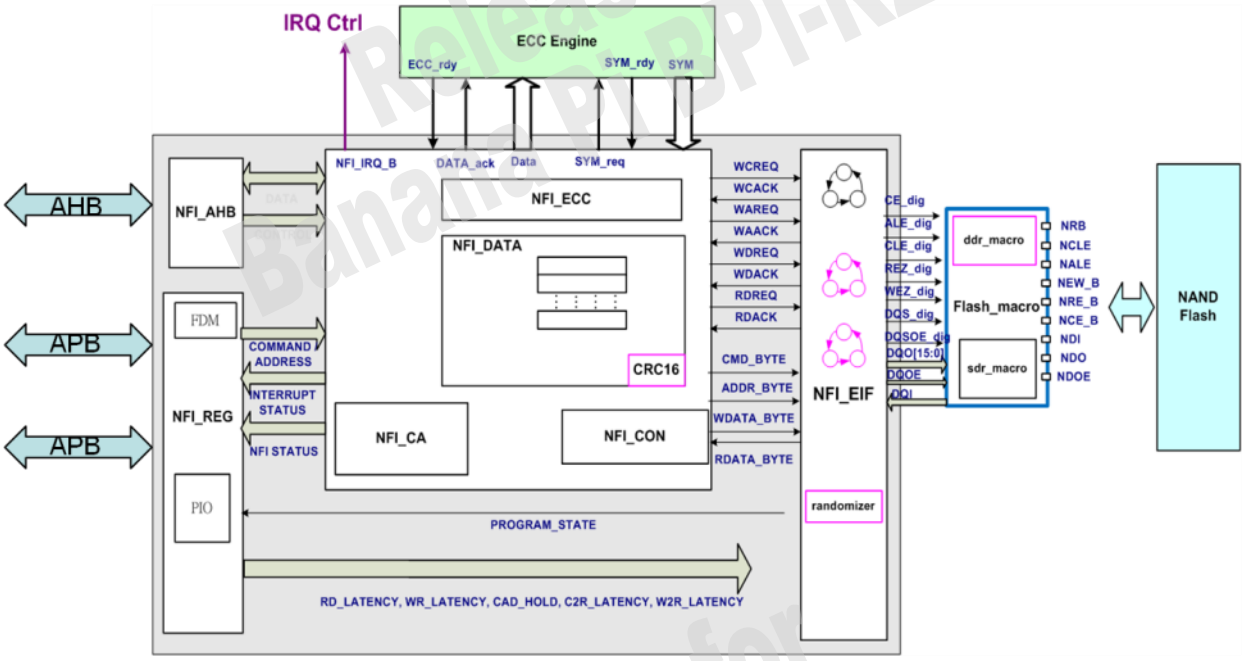


Figure 14-1: Block Diagram of NAND Flash Interface

14.4 NFI Register Definition

Module name: NFI Base address: (+1100D000h)

Address	Name	Width	Register Function
1100D000	NFI_CNFG	16	NFI Configuration
1100D004	NFI_PAGEFMT	16	NFI Page Format Control Register
1100D008	NFI_CON	32	NFI Operation Control Register
1100D00C	NFI_ACCCON	32	NAND Flash Access Timing Control register
1100D010	NFI_INTR_EN	16	NFI Interrupt Enable Register
1100D014	NFI_INTR	16	NFI Interrupt Status Register
1100D020	NFI_CMD	16	NFI Command register
1100D030	NFI_ADDRNOB	16	NFI Address Length Register
1100D034	NFI_COLADDR	32	NFI Column Address Register
1100D038	NFI_ROWADDR	32	NFI Row Address Register
1100D040	NFI_STRDATA	16	NFI Data Transfer Start Trigger Register
1100D044	NFI_CNRNB	16	NFI Check NAND Ready/Busy Register
1100D050	NFI_DATAW	32	NFI Write Data Buffer
1100D054	NFI_DATAR	32	NFI Read Data Buffer
1100D058	NFI_PIO_DIRDY	16	PIO_mode Data Ready Register
1100D060	NFI_STA	32	NFI State
1100D064	NFI_FIFOSTA	16	NFI FIFO Status
1100D070	NFI_ADDRCNTR	32	NFI Page Address Counter Register
1100D080	NFI_STRADDR	32	NFI AHB Start Address Register
1100D084	NFI_BYTELEN	32	NFI DMA Byte Length Register

1100D090	**NFI_CSEL**	16	**NFI device select register**
1100D094	**NFI_IOCON**	32	**NFI IO Control register**
1100D0A0	**NFI_FDM0L**	32	**NFI Least FDM Data for Sector 0 Register**
1100D0A4	**NFI_FDM0M**	32	**NFI Most FDM Data for Sector 0 Register**
1100D0A8	**NFI_FDM1L**	32	**NFI Least FDM Data for Sector 1 Register**
1100D0AC	**NFI_FDM1M**	32	**NFI Most FDM Data for Sector 1 Register**
1100D0B0	**NFI_FDM2L**	32	**NFI Least FDM Data for Sector 2 Register**
1100D0B4	**NFI_FDM2M**	32	**NFI Most FDM Data for Sector 2 Register**
1100D0B8	**NFI_FDM3L**	32	**NFI Least FDM Data for Sector 3 Register**
1100D0BC	**NFI_FDM3M**	32	**NFI Most FDM Data for Sector 3 Register**
1100D0C0	**NFI_FDM4L**	32	**NFI Least FDM Data for Sector 4 Register**
1100D0C4	**NFI_FDM4M**	32	**NFI Most FDM Data for Sector 4 Register**
1100D0C8	**NFI_FDM5L**	32	**NFI Least FDM Data for Sector 5 Register**
1100D0CC	**NFI_FDM5M**	32	**NFI Most FDM Data for Sector 5 Register**
1100D0D0	**NFI_FDM6L**	32	**NFI Least FDM Data for Sector 6 Register**
1100D0D4	**NFI_FDM6M**	32	**NFI Most FDM Data for Sector 6 Register**
1100D0D8	**NFI_FDM7L**	32	**NFI Least FDM Data for Sector 7 Register**
1100D0DC	**NFI_FDM7M**	32	**NFI Most FDM Data for Sector 7 Register**
1100D0E0	**NFI_FDM8L**	32	**NFI Least FDM Data for Sector 8 Register**
1100D0E4	**NFI_FDM8M**	32	**NFI Most FDM Data for Sector 8 Register**
1100D0E8	**NFI_FDM9L**	32	**NFI Least FDM Data for Sector 9 Register**
1100D0EC	**NFI_FDM9M**	32	**NFI Most FDM Data for Sector 9 Register**
1100D0F0	**NFI_FDMAL**	32	**NFI Least FDM Data for Sector A Register**
1100D0F4	**NFI_FDMAM**	32	**NFI Most FDM Data for Sector A Register**
1100D0F8	**NFI_FDMBL**	32	**NFI Least FDM Data for Sector B Register**
1100D0FC	**NFI_FDMBM**	32	**NFI Most FDM Data for Sector B Register**
1100D100	**NFI_FDMCL**	32	**NFI Least FDM Data for Sector C Register**
1100D104	**NFI_FDMCM**	32	**NFI Most FDM Data for Sector C Register**
1100D108	**NFI_FDMDL**	32	**NFI Least FDM Data for Sector D Register**
1100D10C	**NFI_FDMDM**	32	**NFI Most FDM Data for Sector D Register**
1100D110	**NFI_FDMEL**	32	**NFI Least FDM Data for Sector E Register**
1100D114	**NFI_FDMEM**	32	**NFI Most FDM Data for Sector E Register**
1100D118	**NFI_FDMFL**	32	**NFI Least FDM Data for Sector F Register**
1100D11C	**NFI_FDMFM**	32	**NFI Most FDM Data for Sector F Register**
1100D120	**NFI_CRC01**	32	**NFI CRC Data for Sector 0 and 1 Register**
1100D124	**NFI_CRC23**	32	**NFI CRC Data for Sector 2 and 3 Register**
1100D128	**NFI_CRC45**	32	**NFI CRC Data for Sector 4 and 5 Register**
1100D12C	**NFI_CRC67**	32	**NFI CRC Data for Sector 6 and 7 Register**
1100D130	**NFI_CRC89**	32	**NFI CRC Data for Sector 8 and 9 Register**
1100D134	**NFI_CRCAB**	32	**NFI CRC Data for Sector A and B Register**
1100D138	**NFI_CRCCD**	32	**NFI CRC Data for Sector C and D Register**
1100D13C	**NFI_CRCEF**	32	**NFI CRC Data for Sector E and F Register**
1100D190	**NFI_FIFODATA0**	32	**NFI FIFO Content Data 0**
1100D194	**NFI_FIFODATA1**	32	**NFI FIFO Content Data 1**
1100D198	**NFI_FIFODATA2**	32	**NFI FIFO Content Data 2**
1100D19C	**NFI_FIFODATA3**	32	**NFI FIFO Content Data 3**
1100D200	**NFI_MCON**	16	**NFI LCD Monitor Control Register**

1100D204	NFI_TOTALCNT	32	NFI LCD Monitor Total Cycle Count
1100D208	NFI_RQCNT	32	NFI LCD Monitor Request Cycle Count
1100D20C	NFI_ACCNT	32	NFI LCD Monitor Access Cycle Count
1100D220	NFI_DEBUG_CON1	16	NFI Debug register
1100D224	NFI_MASTER_STA	16	NFI Master Status
1100D228	NFI_MASTER_RST	32	pad macro soft reset
1100D22C	NFI_SECCUS_SIZE	32	Enable bit for customized size for each sector
1100D230	NFI_SPIADDRCNTR	32	NFI AHB Start Address Register
1100D234	NFI_SPIBYTELEN	32	NFI DMA Byte Length Register
1100D238	NFI_RANDOM_CNFG	32	Randomizer Config
1100D23C	NFI_EMPTY_THRESH	32	Empty threshold setting
1100D240	NFI_NAND_TYPE_CNFG	32	toggle/synchronous/asynchronous interface and timing mode setting
1100D244	NFI_ACCCON1	32	NFI access timing setting1
1100D248	NFI_DELAY_CTRL	32	DQS and DQ delay control
1100D24C	NFI_RANDOM_ENSEED01_TS	32	TOSHIBA encode seed0 seed1
1100D250	NFI_RANDOM_ENSEED2_TS	32	TOSHIBA encode seed2
1100D254	NFI_RANDOM_ENSEED3_TS	32	TOSHIBA encode seed3
1100D258	NFI_RANDOM_ENSEED45_TS	32	TOSHIBA encode seed4 seed5
1100D25C	NFI_RANDOM_ENSEED6_TS	32	TOSHIBA encode seed6
1100D260	NFI_RANDOM_ENSEED7_TS	32	TOSHIBA encode seed7
1100D264	NFI_RANDOM_DESEED01_TS	32	TOSHIBA encode seed0 seed1
1100D268	NFI_RANDOM_DESEED2_TS	32	TOSHIBA encode seed2
1100D26C	NFI_RANDOM_DESEED3_TS	32	TOSHIBA encode seed3
1100D270	NFI_RANDOM_DESEED45_TS	32	TOSHIBA encode seed4 seed5
1100D274	NFI_RANDOM_DESEED6_TS	32	TOSHIBA encode seed6
1100D278	NFI_RANDOM_DESEED7_TS	32	TOSHIBA encode seed7
1100D500	SNF_MAC_CTL	32	Serial nand flash mac mode control
1100D504	SNF_MAC_OUTL	32	Serial nand flash mac mode output data length
1100D508	SNF_MAC_INL	32	Serial nand flash mac mode input data length
1100D50C	SNF_RD_CTL1	32	Serial nand flash read control 1
1100D510	SNF_RD_CTL2	32	Serial nand flash read control 2
1100D514	SNF_RD_CTL3	32	Serial nand flash read control 3
1100D518	SNF_GF_CTL1	32	Serial nand flash get feature control 1
1100D520	SNF_GF_CTL3	32	Serial nand flash mac mode control
1100D524	SNF_PG_CTL1	32	Serial nand flash program control1
1100D528	SNF_PG_CTL2	32	Serial nand flash program control2
1100D52C	SNF_PG_CTL3	32	Serial nand flash program control3
1100D530	SNF_ER_CTL	32	Serial nand flash erase control

1100D534	**SNF_ER_CTL2**	32	**Serial nand flash erase control 2**
1100D538	**SNF_MISC_CTL**	32	**Serial nand flash MISC control**
1100D53C	**SNF_MISC_CTL2**	32	**Serial nand flash MISC control 2**
1100D540	**SNF_DLY_CTL1**	32	**Serial nand flash delay control setting 1**
1100D544	**SNF_DLY_CTL2**	32	**Serial nand flash delay control setting 2**
1100D548	**SNF_DLY_CTL3**	32	**Serial nand flash control setting 3**
1100D54C	**SNF_DLY_CTL4**	32	**Serial nand flash delay control setting 4**
1100D550	**SNF_STA_CTL1**	32	
1100D554	**SNF_STA_CTL2**	32	
1100D558	**SNF_STA_CTL3**	32	
1100D55C	**SNF_SNF_CNFG**	32	**SPI/Parallel NAND Selection**
1100D560	**SNF_DEBUG_SEL**	32	**DEBUG MUX Selection**

1100D000 NFI_CNFG NFI Configuration 0020

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		OP_MODE					AUTO_FMT_EN	HW_ECC_EN	CRC_EN	BYTE_RW	SS_RANDOMIZER_SEL	RESERED_SEC_EN		DMA_BURST_EN	READ_MODE	DMA_MODE
Type		RW					RW	RW	RW	RW	RW	RW		RW	RW	RW
Reset		0	0	0			0	0	0	0	1	0		0	0	0

Bit(s)	Name	Description
14:12	OP_MODE	**The field control the operating process flow of FSM for NFI.** 000b: Idle state. 001b: Read Process. Recommend for basic read operation. 010b: Single Read Process. Recommend for read id and read status. 011b: Program Process. Recommend for basic program operation. 100b: Erase Process. Recommend for basic erase operation. 101b: Reset Process. Recommend for basic reset operation. 110b: Custom Process. Recommend for all advance operation. Others: Reserved
9	AUTO_FMT_EN	**Automatic HW ECC encode or decode enable.** If enabled, the ECC parity from HW ECC engine and FDM data from Register are written automatically to the spare area. If disable, the spare data all comes from PIO register, like DATAR, DATAW, (PIO Mode) or the memory(DMA Mode) as main area data.
8	HW_ECC_EN	**This field is used to enable encoding or decoding operation of HW ECC engine. If the bit is enabled, the data is transferring to ECC engine for encoding and decoding. The ECC Engine should be configured as nfi encoding mode, otherwise the NFI will hang.**

UTO_FMT_EN	HW_ECC_EN	NFI_FUNCTION
	0	DATA,FDM,CRC,Partiy all come from PIO/DMA, ECC interface turn off
	1	DATA,FDM,CRC,Partiy all come from PIO/DMA, ECC interface turn on
	0	DATA come from APB/DMA,FDM come from FDM register,Partiy force 0xff, ECC interface turn off
	1	DATA come from APB/DMA,FDM come from FDM register,Partiy come from ECC, ECC interface turn off

7	CRC_EN	**Enable CRC data auto generate after main data and fdm data .**
6	BYTE_RW	**Enable byte access. The valid bytes read from NFI_DATAR and NFI_DATAW is only DR0 and DW0 if BYTE_RW is enabled.**
5	SS_RANDOMIZER_SEL	1: SAMSANG 0: TOSHIBA
4	RESEED_SEC_EN	**If this bit is enable, HW will reload initial seed to randomizer for each sector, otherwise reload seed for each page** 0: reseed for each page 1: reseed for each sector
2	DMA_BURST_EN	**When NFI is in DMA mode, single and burst(incremental) trans type are used. If start address is not alignment,single trans will auto be issued till address is alignment when DMA_BURST_EN is enabled in DMA mode** 0: DMA burst transaction disable 1: DMA burst transaction enable
1	READ_MODE	**This field is used to control the activity of read or write transfer.** 0: write operation of DMA or PIO. 1: read operation of DMA or PIO.
0	DMA_MODE	**This field is used to control the Operation mode.** 0: PIO mode. All data (include read or write) move by MCU through APB access. 1: DMA mode. All data (include read or write) move by HW automation through AHB bus.

1100D004 NFI_PAGEFMT NFI Page Format Control Register 0004

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM_ECC_NUM				FDM_NUM				SPARE_SIZE				DBYTE_EN	SEC_SEL_512	DATA_AREA_PAGE_SIZE	
Type	RW				RW				RW				RW	RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

Bit(s)	Name	Description
15:12	FDM_ECC_NUM	**The number of each FDM data for HW ECC protection. The valid number of bytes ranges are from 0 to 8.**
11:8	FDM_NUM	**The FDM data number for each spare area. The valid number of bytes are from 0 to 8.**
7:4	SPARE_SIZE	**The fields represents the spare size value for each sector . If sec_size is 1K(sel_sec_512 is 0), spare size will be doubled.** 0000b: 16Bytes 0001b: 26Bytes 0010b: 27Bytes 0011b: 28Bytes 0100b: 32Bytes 0101b: 36Bytes 0110b: 40Bytes 0111b: 44Bytes 1000b: 48Bytes 1001b: 49Bytes 1010b: 50Bytes 1011b: 51Bytes 1100b: 52Bytes 1101b: 62Bytes 1110b: 63Bytes 1111b: 64Bytes
3	DBYTE_EN	**16 bits I/O bus interface enable.**
2	SEC_SEL_512	**sec_size select.** 1: sec size 512B 0: sec size 1024B
1:0	DATA_AREA_PAGE_SIZE	**Page Size Of Data area.(the real page size will including data area and (spare_size* sec_num) bytes spare area).The field specifies the size of one page**

for the device. Some most widely used page size are supported.
00b: The page size is 512B if sel_sec_size is 1, otherwise page size is 2kB
01b: The page size is 2kB if sel_sec_size is 1,otherwise page size is 4kB .
10b: The page size is 4kB if sel_sec_size is 1, otherwise page size is 8kB .
11b: Reserved if sel_sec_size is 1, otherwise page size is 16kB.

1100D008 NFI_CON NFI Operation Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SEC_NUM[4:4]
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEC_NUM[3:0]						BWR	BRD	NOB			SRD			NFI_RST	FIFO_FLUSH
Type	RW						RW	RW	RW			WO			WO	WO
Reset	0	0	0	0			0	0	0	0	0	0			0	0

Bit(s)	Name	Description
16:12	SEC_NUM	**The field represents the sector number to be retrieved from the device or DMA Master. The valid number ranges from 1 to 16.**
9	BWR	**Burst write mode. Setting to be logic-1 enables the data burst write operation.**
8	BRD	**Burst read mode. Setting this field to be logic-1 enables the data read operation. The NFI core will issue read cycles to retrieve data from the device when the data FIFO is not full or the device is not in the busy state. The NFI core supports consecutive page reading.**
7:5	NOB	**The field represents the number of bytes to be retrieved from the device in single mode, and the number of bytes per APB transaction in both single and burst mode. If device is 16-bit IO or sync/toggle interface, the read bytes number will double** 0: Read 8 bytes from the device. 1: Read 1 byte from the device. 2: Read 2 bytes from the device. 3: Read 3 bytes from the device. 4: Read 4 bytes from the device. 5: Read 5 byte from the device. 6: Read 6 bytes from the device. 7: Read 7 bytes from the device.
4	SRD	**Setting to be logic-1 initializes the one-shot data read operation. It's mainly used for read ID and read status command, which requires no more than 4 read cycles to retrieve data from the device. It used when FIFO is empty or after reset nficore**
1	NFI_RST	**Reset the state machine, data FIFO (0x0000) and FDM data (0xffff)**
0	FIFO_FLUSH	**Flush the data FIFO.**

1100D00C NFI_ACCCON NAND Flash Access Timing Control register F3FFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	POECS				PRECS						C2R					
Type	RW				RW						RW					
Reset	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	W2R				WH				WST				RLT			
Type	RW				RW				RW				RW			
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description

31:28	POECS	**The field represents the minimum required time for CS post-pulling down after the access to device.** Minimum required time = PRECS[1:0] + PRECS[2]*8 + PRECS[3]*64 (T)
27:22	PRECS	**The field represents the minimum required time for CS pre-pulling down before any access to device.** Minimum required time = PRECS[1:0] + PRECS[3:2]*8 + PRECS[5:4]*128 (T)
21:16	C2R	**The field represents the minimum required time from NCEB low to NREB low. It's in unit of 2T.** Minimum required time = C2R[5:0]*2 + 1 (T)
15:12	W2R	**The field represents the minimum required time from NWEB high to NREB low. It's in unit of 2T. So the actual time ranges from 0T to 30T in step of 2T.** Minimum required time = W2R[3:0]*2 + 1 (T)
11:8	WH	**Write-enable hold-time.** The field specifies the hold time of NALE, NCLE, NCEB signals relative to the rising edge of NWEB. This field is associated with WST to expand the write cycle time, and is associated with RLT to expand the read cycle time.
7:4	WST	**Write Wait State** The field specifies the wait states to be inserted to meet the requirement of the pulse width of the NWEB signal. 00b: No wait state. 01b: 1T wait state. 10b: 2T wait state. 11b: 3T wait state.
3:0	RLT	**Read Latency Time** The field specifies how many wait states to be inserted to meet the requirement of the read access time for the device. 00b: No wait state. 01b: 1T wait state. 10b: 2T wait state. 11b: 3T wait state.

1100D010 NFI_INTR_EN NFI Interrupt Enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					AUTO_BLKER_INTR_EN	AUTO_READ_INTR_EN	AUTO_PROGRAM_INTR_EN	CUSTOM_READ_INTR_EN	CUSTOM_PROGRAM_INTR_EN	AHB_DONE_EN	ACCESS_LOCK_EN	BUSY_RETURN_EN	ERASE_DONE_EN	RESET_DONE_EN	WR_DONE_EN	RD_DONE_EN
Type					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11	AUTO_BLKER_INTR_EN	**The done interrupt enable for spi auto block earase**
10	AUTO_READ_INTR_EN	**The done interrupt enable for spi auto read**
9	AUTO_PROGRAM_INTR_EN	**The done interrupt enable for spi auto program**
8	CUSTOM_READ_INTR_EN	**The done interrupt enable for spi custom read**
7	CUSTOM_PROGRAM_INTR_EN	**The done interrupt enable for spi custom program**
6	AHB_DONE_EN	**The done interrupt enable for DMA mode.**
5	ACCESS_LOCK_EN	
4	BUSY_RETURN_EN	**The busy return interrupt enable.**
3	ERASE_DONE_EN	**The erase completion interrupt enable.**
2	RESET_DONE_EN	**The reset completion interrupt enable.**
1	WR_DONE_EN	**The single page write completion interrupt enable.**

0	RD_DONE_EN	**The single page read completion interrupt enable.**

1100D014 NFI_INTR NFI Interrupt Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					AUTO_BLKER_INTR	AUTO_READ_INTR	AUTO_PROGRAM_INTR	CUSTOM_READ_INTR	CUSTOM_PROGRAM_INTR	AHB_DONE	ACCESS_LOCK	BUSY_RETURN	ERASE_DONE	RESET_DONE	WR_DONE	RD_DONE
Type					RC	RC	RC	RC	RC	RC	RC	RC	RC	RC	RC	RC
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11	AUTO_BLKER_INTR	**Indicates that the spi auto erase is completed.**
10	AUTO_READ_INTR	**Indicates that the spi auto read is completed.**
9	AUTO_PROGRAM_INTR	**Indicates that the spi auto program is completed.**
8	CUSTOM_READ_INTR	**Indicates that the spi custom read is completed.**
7	CUSTOM_PROGRAM_INTR	**Indicates that the spi custom program is completed.**
6	AHB_DONE	**Indicates that the AHB operation is completed.**
5	ACCESS_LOCK	
4	BUSY_RETURN	**Indicates that the device state returns from busy by inspecting the R/B# pin.**
3	ERASE_DONE	**Indicates that the erase operation is completed.**
2	RESET_DONE	**Indicates that the reset operation is completed.**
1	WR_DONE	**Indicates that the write operation is completed.**
0	RD_DONE	**Indicates that the single page read operation is completed.**

1100D020 NFI_CMD NFI Command register 0045

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CMD							
Type									RW							
Reset									0	1	0	0	0	1	0	1

Bit(s)	Name	Description
7:0	CMD	**Command word.**

1100D030 NFI_ADDRNOB NFI Address Length Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										ROW_ADDR_NOB				COL_ADDR_NOB		
Type										RW				RW		
Reset										0	0	0		0	0	0

Bit(s)	Name	Description
6:4	ROW_ADDR_NOB	**Number of bytes for the row address**
2:0	COL_ADDR_NOB	**Number of bytes for the column address**

1100D034 NFI_COLADDR NFI Column Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	COL_ADDR3								COL_ADDR2							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	COL_ADDR1								COL_ADDR0							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	COL_ADDR3	**The 3-th column address byte.**
23:16	COL_ADDR2	**The 2-th column address byte.**
15:8	COL_ADDR1	**The 1-th column address byte.**
7:0	COL_ADDR0	**The 0-th column address byte.**

1100D038 NFI_ROWADDR NFI Row Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ROW_ADDR3								ROW_ADDR2							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ROW_ADDR1								ROW_ADDR0							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	ROW_ADDR3	**The 3-th row address byte.**
23:16	ROW_ADDR2	**The 2-th row address byte.**
15:8	ROW_ADDR1	**The 1-th row address byte.**
7:0	ROW_ADDR0	**The 0-th row address byte.**

1100D040 NFI_STRDATA NFI Data Transfer Start Trigger Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																STR_DATA
Type																WO
Reset																0

Bit(s)	Name	Description
0	STR_DATA	**This signal triggers the data transfer for read or write. It only takes effect as custom operation mode**

1100D044 NFI_CNRNB NFI Check NAND Ready/Busy Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CB2R_TIME							STR_CNRNB
Type									RW							WO
Reset									0	0	0	0				0

Bit(s)	Name	Description
7:4	CB2R_TIME	**This time-out registers for polling the NAND busy/ready signal. The unit is 16T clock cycles. The clock rate is 61.44MHz in normal mode. It will be slow down after enable HW DCM mode.**
0	STR_CNRNB	**This signal triggers NFI to poll the status the NAND busy/ready signal after CB2R_TIME*16 cycles. This function is used to avoid the fail function of "BUSY2READY" status or "BUSY_RETURN" interrupt when NAND is operating at very low frequency(<7MHz). If NAND is operating in lower frequency, the sampling for the event, NAND busy/ready signal from low to high, may be failed and NFI will be hanged in busy state. This signal is a time-out register to check the NAND status. The results will be report to "BUSY2READY" status and "BUSY_RETURN" interrupt.**

1100D050 NFI_DATAW NFI Write Data Buffer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DW3								DW2							
Type	WO								WO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DW1								DW0							
Type	WO								WO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DW3	**Write data byte 3.**
23:16	DW2	**Write data byte 2.**
15:8	DW1	**Write data byte 1.**
7:0	DW0	**Write data byte 0.**

1100D054 NFI_DATAR NFI Read Data Buffer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DR3								DR2							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DR1								DR0							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	DR3	**Read data byte 3.**
23:16	DR2	**Read data byte 2.**
15:8	DR1	**Read data byte 1.**
7:0	DR0	**Read data byte 0.**

1100D058 NFI_PIO_DIRDY PIO_mode Data Ready Register 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																PIO_DI_RDY
Type																RO
Reset																1

Bit(s)	Name	Description
0	PIO_DI_RDY	**indicates the PIO mode is ready for read data in read mode and ready for write data in write mode.** 0: NFI_DATAR and NFI_DATAW should not be read or write (not ready). 1: NFI is ready for reading data in ready mode and writing data in write mode.

1100D060 NFI_STA NFI State 00001020

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NAND_FSM_TYPE			NAND_FSM									NFI_FSM			
Type	RO			RO									RO			
Reset	0	0	0	0	0	0	0	0	0				0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				READ_EMPTY			BUSY2READY	BUSY			FLASH_MACRO_IDLE	ACCESS_LOCK	DATAW	DATAR	ADDR	CMD
Type				RO			RO	RO			RO	RO	RO	RO	RO	RO
Reset				1			0	0			1	0	0	0	0	0

Bit(s)	Name	Description
31:30	NAND_FSM_TYPE	**The field represents the state of NAND interface FSM type: Async state,Sync state Toggle state.** 00b: Async_FSM 01b: Sync_FSM 10b: Toggle_FSM 11b: Reserved
29:23	NAND_FSM	**The field represents the state of NAND interface FSM.** 000000b: IDLE. idle. 111000b: PRE_CS. Pre CS state. 001001b: CMD_WRST. command write set up 001010b: CMD_WR. Command write enable. 001011b: CMD_WRHD. Command write hold. 001000b: CMD_WRRDY 0100001b: ADDR_WRST. Address write set up 0100010b: ADDR_WR. Address write enable 0100011b: ADDR_WRHD. Address write hold 0101111b: ADDR_WRRDY. 0111111b: CA2DEXT. Command address write extension. 1001100b: DATA_RDST. Data read set up. 1000010b: DATA_RD. Data read enable. 1000011b: DATA_RDHD. Data read hold. 1101100b: DATA_WRST. Data write set up. 1100010b: DATA_WR. Data write enable. 1100011b: DATA_WRHD. Data write hold. Others: Reserved
19:16	NFI_FSM	**The field represents the state of NFI internal FSM.** 0000b: idle. 0001b: reset. Reset command to ready 0010b: read busy. 0011b: read data. 0100b: program busy 0101b: program data. Input data command to program command 1000b: erase busy. Erase command to ready 1001b: erase data. Erase command 1 to erase command 2 1111b: custom mode 1110b: custom mode for data access Others: Reserved
12	READ_EMPTY	**Empty page indication during read operation, include all data, FDM and parity for all sectors**
9	BUSY2READY	**It's read-only. This signal indicates NAND from busy to ready state and it will be**

		reset after nfi_reset or write command/address.
8	BUSY	**Synchronized busy signal from the NAND flash. It's read-only. This signal is sampled from NFI**
5	FLASH_MACRO_IDLE	**This signal indicates flash macro is in idle.** Note: Only flash macro is idle, sw reset or nand interface change can be issued. 1: idle 0: non idle
4	ACCESS_LOCK	**The access range is locked for erase or program .**
3	DATAW	**The NFI core is in data write mode.**
2	DATAR	**The NFI core is in data read mode.**
1	ADDR	**The NFI core is in address mode.**
0	CMD	**The NFI core is in command mode.**

1100D064 NFI_FIFOSTA NFI FIFO Status 4040

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WR_FULL	WR_EMPTY		WR_REMAIN					RD_FULL	RD_EMPTY		RD_REMAIN				
Type	RO	RO		RO					RO	RO		RO				
Reset	0	1		0	0	0	0	0	0	1		0	0	0	0	0

Bit(s)	Name	Description
15	WR_FULL	**Data FIFO full in burst write mode.**
14	WR_EMPTY	**Data FIFO empty in burst write mode.**
12:8	WR_REMAIN	**Data FIFO remaining byte number in burst write mode.**
7	RD_FULL	**Data FIFO full in burst read mode.**
6	RD_EMPTY	**Data FIFO empty in burst read mode.**
4:0	RD_REMAIN	**Data FIFO remaining byte number in burst read mode.**

1100D070 NFI_ADDRCNTR NFI Page Address Counter Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SEC_CNTR[4:4]
Type																RO
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SEC_CNTR[3:0]				SEC_ADDR											
Type	RO				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16:12	SEC_CNTR	**The sector count.**
11:0	SEC_ADDR	**The address count of 512 main data and spare data for each sector.**

1100D080 NFI_STRADDR NFI AHB Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	STR_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	STR_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	STR_ADDR	**The start address of EMI for both read or write in DMA mode.** If start address of any sector data is not 4-byte aligned, the transfer will be automatically split into byte and word transaction by NFI DMA. Non 4-byte aligned data will be transferred in single-byte transaction. Non 16-byte aligned data will be transferred in single-word transaction. 16-byte aligned data will be transferred by 4 word incrementing bust if the NFI_CNFG->DMA_BURST_EN is enabled.

1100D084 NFI_BYTELEN NFI DMA Byte Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																BUS_SEC_CNTR[4:4]
Type																RO
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BUS_SEC_CNTR[3:0]				BUS_SEC_ADDR											
Type	RO				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16:12	BUS_SEC_CNTR	**The sector count.**
11:0	BUS_SEC_ADDR	**The address count of 512 main data and spare data for each sector.**

1100D090 NFI_CSEL NFI device select register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																CSEL
Type																RW
Reset																0

Bit(s)	Name	Description
0	CSEL	**Chip select. The value defaults to 0.** 0: Device 1 is selected. 1: Device 2 is selected.

1100D094 NFI_IOCON NFI IO Control register 00000006

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									BRSTN					L2NW	L2NR	NLD_PD
Type									RW					RW	RW	RW
Reset									0	0	0	0		1	1	0

Bit(s)	Name	Description
7:4	BRSTN	**Maximum Burst Number for NAND read and writes. The unit is number of byte (8bits I/O) or double byte (16bits I/O)**
2	L2NW	**Enable 1T latency for the arbitration from LCD to NAND write operation, this is used to prevent bus contention between chip, NAND flash and LCD device.**
1	L2NR	**Enable 1T latency for the arbitration from LCD to NAND read operation, this is used to prevent bus contention between chip, NAND flash and LCD device.**
0	NLD_PD	**data bus pull down when no use.** 0: disable. 1: enable.

1100D0A0 NFI_FDM0L NFI Least FDM Data for Sector 0 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM0_3								FDM0_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM0_1								FDM0_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM0_3	**The 3-th FDM byte data for sector 0.**
23:16	FDM0_2	**The 2-th FDM byte data for sector 0.**
15:8	FDM0_1	**The 1-th FDM byte data for sector 0.**
7:0	FDM0_0	**The 0-th FDM byte data for sector 0.**

1100D0A4 NFI_FDM0M NFI Most FDM Data for Sector 0 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM0_7								FDM0_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM0_5								FDM0_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM0_7	**The 3-th FDM byte data for sector 0.**
23:16	FDM0_6	**The 2-th FDM byte data for sector 0.**
15:8	FDM0_5	**The 1-th FDM byte data for sector 0.**
7:0	FDM0_4	**The 0-th FDM byte data for sector 0.**

1100D0A8 NFI_FDM1L NFI Least FDM Data for Sector 1 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM1_3								FDM1_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM1_1								FDM1_0							
Type	RW								RW							

Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM1_3	**The 3-th FDM byte data for sector 1.**
23:16	FDM1_2	**The 2-th FDM byte data for sector 1.**
15:8	FDM1_1	**The 1-th FDM byte data for sector 1.**
7:0	FDM1_0	**The 0-th FDM byte data for sector 1.**

1100D0AC NFI_FDM1M NFI Most FDM Data for Sector 1 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM1_7								FDM1_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM1_5								FDM1_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM1_7	**The 7-th FDM byte data for sector 1.**
23:16	FDM1_6	**The 6-th FDM byte data for sector 1.**
15:8	FDM1_5	**The 5-th FDM byte data for sector 1.**
7:0	FDM1_4	**The 4-th FDM byte data for sector 1.**

1100D0B0 NFI_FDM2L NFI Least FDM Data for Sector 2 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM2_3								FDM2_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM2_1								FDM2_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM2_3	**The 3-th FDM byte data for sector 2.**
23:16	FDM2_2	**The 2-th FDM byte data for sector 2.**
15:8	FDM2_1	**The 1-th FDM byte data for sector 2.**
7:0	FDM2_0	**The 0-th FDM byte data for sector 2.**

1100D0B4 NFI_FDM2M NFI Most FDM Data for Sector 2 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM2_7								FDM2_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM2_5								FDM2_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM2_7	**The 7-th FDM byte data for sector 2.**
23:16	FDM2_6	**The 6-th FDM byte data for sector 2.**
15:8	FDM2_5	**The 5-th FDM byte data for sector 2.**
7:0	FDM2_4	**The 4-th FDM byte data for sector 2.**

1100D0B8 NFI_FDM3L NFI Least FDM Data for Sector 3 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM3_3								FDM3_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM3_1								FDM3_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM3_3	**The 3-th FDM byte data for sector 3.**
23:16	FDM3_2	**The 2-th FDM byte data for sector 3.**
15:8	FDM3_1	**The 1-th FDM byte data for sector 3.**
7:0	FDM3_0	**The 0-th FDM byte data for sector 3.**

1100D0BC NFI_FDM3M NFI Most FDM Data for Sector 3 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM3_7								FDM3_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM3_5								FDM3_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM3_7	**The 4-th FDM byte data for sector 3.**
23:16	FDM3_6	**The 6-th FDM byte data for sector 3.**
15:8	FDM3_5	**The 5-th FDM byte data for sector 3.**
7:0	FDM3_4	**The 4-th FDM byte data for sector 3.**

1100D0C0 NFI_FDM4L NFI Least FDM Data for Sector 4 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM4_3								FDM4_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM4_1								FDM4_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM4_3	**The 3-th FDM byte data for sector 4.**
23:16	FDM4_2	**The 2-th FDM byte data for sector 4.**
15:8	FDM4_1	**The 1-th FDM byte data for sector 4.**
7:0	FDM4_0	**The 0-th FDM byte data for sector 4.**

1100D0C4 NFI_FDM4M NFI Most FDM Data for Sector 4 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM4_7								FDM4_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM4_5								FDM4_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM4_7	**The 7-th FDM byte data for sector 4.**
23:16	FDM4_6	**The 6-th FDM byte data for sector 4.**
15:8	FDM4_5	**The 5-th FDM byte data for sector 4.**
7:0	FDM4_4	**The 4-th FDM byte data for sector 4.**

1100D0C8 NFI_FDM5L NFI Least FDM Data for Sector 5 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM5_3								FDM5_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM5_1								FDM5_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM5_3	**The 3-th FDM byte data for sector 5.**
23:16	FDM5_2	**The 2-th FDM byte data for sector 5.**
15:8	FDM5_1	**The 1-th FDM byte data for sector 5.**
7:0	FDM5_0	**The 0-th FDM byte data for sector 5.**

1100D0CC NFI_FDM5M NFI Most FDM Data for Sector 5 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM5_7								FDM5_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM5_5								FDM5_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description

31:24	FDM5_7	**The 7-th FDM byte data for sector 5.**
23:16	FDM5_6	**The 6-th FDM byte data for sector 5.**
15:8	FDM5_5	**The 5-th FDM byte data for sector 5.**
7:0	FDM5_4	**The 4-th FDM byte data for sector 5.**

1100D0D0 NFI_FDM6L NFI Least FDM Data for Sector 6 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM6_3								FDM6_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM6_1								FDM6_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM6_3	**The 3-th FDM byte data for sector 6.**
23:16	FDM6_2	**The 2-th FDM byte data for sector 6.**
15:8	FDM6_1	**The 1-th FDM byte data for sector 6.**
7:0	FDM6_0	**The 0-th FDM byte data for sector 6.**

1100D0D4 NFI_FDM6M NFI Most FDM Data for Sector 6 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM6_7								FDM6_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM6_5								FDM6_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM6_7	**The 7-th FDM byte data for sector 6.**
23:16	FDM6_6	**The 6-th FDM byte data for sector 6.**
15:8	FDM6_5	**The 5-th FDM byte data for sector 6.**
7:0	FDM6_4	**The 4-th FDM byte data for sector 6.**

1100D0D8 NFI_FDM7L NFI Least FDM Data for Sector 7 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM7_3								FDM7_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM7_1								FDM7_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM7_3	**The 3-th FDM byte data for sector 7.**

23:16	FDM7_2	**The 2-th FDM byte data for sector 7.**
15:8	FDM7_1	**The 1-th FDM byte data for sector 7.**
7:0	FDM7_0	**The 0-th FDM byte data for sector 7.**

1100D0DC NFI_FDM7M NFI Most FDM Data for Sector 7 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM7_7								FDM7_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM7_5								FDM7_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM7_7	**The 7-th FDM byte data for sector 7.**
23:16	FDM7_6	**The 6-th FDM byte data for sector 7.**
15:8	FDM7_5	**The 5-th FDM byte data for sector 7**
7:0	FDM7_4	**The 4-th FDM byte data for sector 7.**

1100D0E0 NFI_FDM8L NFI Least FDM Data for Sector 8 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM8_3								FDM8_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM8_1								FDM8_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM8_3	**The 3-th FDM byte data for sector 8.**
23:16	FDM8_2	**The 2-th FDM byte data for sector 8.**
15:8	FDM8_1	**The 1-th FDM byte data for sector 8.**
7:0	FDM8_0	**The 0-th FDM byte data for sector 8.**

1100D0E4 NFI_FDM8M NFI Most FDM Data for Sector 8 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM8_7								FDM8_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM8_5								FDM8_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM8_7	**The 7-th FDM byte data for sector 8.**
23:16	FDM8_6	**The 6-th FDM byte data for sector 8.**

15:8	FDM8_5	**The 5-th FDM byte data for sector 8.**
7:0	FDM8_4	**The 4-th FDM byte data for sector 8.**

1100D0E8 NFI_FDM9L NFI Least FDM Data for Sector 9 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM9_3								FDM9_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM9_1								FDM9_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM9_3	**The 3-th FDM byte data for sector 9.**
23:16	FDM9_2	**The 2-th FDM byte data for sector 9.**
15:8	FDM9_1	**The 1-th FDM byte data for sector 9.**
7:0	FDM9_0	**The 0-th FDM byte data for sector 9.**

1100D0EC NFI_FDM9M NFI Most FDM Data for Sector 9 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDM9_7								FDM9_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDM9_5								FDM9_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDM9_7	**The 7-th FDM byte data for sector 9.**
23:16	FDM9_6	**The 6-th FDM byte data for sector 9.**
15:8	FDM9_5	**The 5-th FDM byte data for sector 9.**
7:0	FDM9_4	**The 4-th FDM byte data for sector 9.**

1100D0F0 NFI_FDMAL NFI Least FDM Data for Sector A Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMA_3								FDMA_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMA_1								FDMA_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMA_3	**The 3-th FDM byte data for sector A.**
23:16	FDMA_2	**The 2-th FDM byte data for sector A.**
15:8	FDMA_1	**The 1-th FDM byte data for sector A.**

7:0	FDMA_0	**The 0-th FDM byte data for sector A.**

1100D0F4 NFI_FDMAM NFI Most FDM Data for Sector A Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMA_7								FDMA_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMA_5								FDMA_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMA_7	**The 7-th FDM byte data for sector A.**
23:16	FDMA_6	**The 6-th FDM byte data for sector A.**
15:8	FDMA_5	**The 5-th FDM byte data for sector A.**
7:0	FDMA_4	**The 4-th FDM byte data for sector A.**

1100D0F8 NFI_FDMBL NFI Least FDM Data for Sector B Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMB_3								FDMB_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMB_1								FDMB_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMB_3	**The 3-th FDM byte data for sector B.**
23:16	FDMB_2	**The 2-th FDM byte data for sector B.**
15:8	FDMB_1	**The 1-th FDM byte data for sector B.**
7:0	FDMB_0	**The 0-th FDM byte data for sector B.**

1100D0FC NFI_FDMBM NFI Most FDM Data for Sector B Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMB_7								FDMB_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMB_5								FDMB_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMB_7	**The 7-th FDM byte data for sector B.**
23:16	FDMB_6	**The 6-th FDM byte data for sector B.**
15:8	FDMB_5	**The 5-th FDM byte data for sector B.**
7:0	FDMB_4	**The 4-th FDM byte data for sector B.**

1100D100 NFI_FDMCL NFI Least FDM Data for Sector C Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMC_3								FDMC_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMC_1								FDMC_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMC_3	**The 3-th FDM byte data for sector C.**
23:16	FDMC_2	**The 2-th FDM byte data for sector C.**
15:8	FDMC_1	**The 1-th FDM byte data for sector C.**
7:0	FDMC_0	**The 0-th FDM byte data for sector C.**

1100D104 NFI_FDMCM NFI Most FDM Data for Sector C Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMC_7								FDMC_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMC_5								FDMC_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMC_7	**The 7-th FDM byte data for sector C.**
23:16	FDMC_6	**The 6-th FDM byte data for sector C.**
15:8	FDMC_5	**The 5-th FDM byte data for sector C.**
7:0	FDMC_4	**The 4-th FDM byte data for sector C.**

1100D108 NFI_FDMDL NFI Least FDM Data for Sector D Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMD_3								FDMD_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMD_1								FDMD_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMD_3	**The 3-th FDM byte data for sector D.**
23:16	FDMD_2	**The 2-th FDM byte data for sector D.**
15:8	FDMD_1	**The 1-th FDM byte data for sector D.**
7:0	FDMD_0	**The 0-th FDM byte data for sector D.**

1100D10C NFI_FDMDM NFI Most FDM Data for Sector D Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMD_7								FDMD_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMD_5								FDMD_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMD_7	**The 7-th FDM byte data for sector D.**
23:16	FDMD_6	**The 6-th FDM byte data for sector D.**
15:8	FDMD_5	**The 5-th FDM byte data for sector D.**
7:0	FDMD_4	**The 4-th FDM byte data for sector D.**

1100D110 NFI_FDMEL NFI Least FDM Data for Sector E Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDME_3								FDME_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDME_1								FDME_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDME_3	**The 3-th FDM byte data for sector E.**
23:16	FDME_2	**The 2-th FDM byte data for sector E.**
15:8	FDME_1	**The 1-th FDM byte data for sector E.**
7:0	FDME_0	**The 0-th FDM byte data for sector E.**

1100D114 NFI_FDMEM NFI Most FDM Data for Sector E Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDME_7								FDME_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDME_5								FDME_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDME_7	**The 7-th FDM byte data for sector E.**
23:16	FDME_6	**The 6-th FDM byte data for sector E.**
15:8	FDME_5	**The 5-th FDM byte data for sector E.**
7:0	FDME_4	**The 4-th FDM byte data for sector E.**

1100D118 NFI_FDMFL NFI Least FDM Data for Sector F Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMF_3								FDMF_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMF_1								FDMF_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMF_3	**The 3-th FDM byte data for sector F.**
23:16	FDMF_2	**The 2-th FDM byte data for sector F.**
15:8	FDMF_1	**The 1-th FDM byte data for sector F.**
7:0	FDMF_0	**The 0-th FDM byte data for sector F.**

1100D11C NFI_FDMFM NFI Most FDM Data for Sector F Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMF_7								FDMF_6							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMF_5								FDMF_4							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMF_7	**The 7-th FDM byte data for sector F.**
23:16	FDMF_6	**The 6-th FDM byte data for sector F.**
15:8	FDMF_5	**The 5-th FDM byte data for sector F.**
7:0	FDMF_4	**The 4-th FDM byte data for sector F.**

1100D0F0 NFI_FDMAL NFI Least FDM Data for Sector A Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FDMA_3								FDMA_2							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FDMA_1								FDMA_0							
Type	RW								RW							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	FDMA_3	**The 3-th FDM byte data for sector A.**
23:16	FDMA_2	**The 2-th FDM byte data for sector A.**
15:8	FDMA_1	**The 1-th FDM byte data for sector A.**
7:0	FDMA_0	**The 0-th FDM byte data for sector A.**

1100D120 NFI_CRC01 NFI CRC Data for Sector 0 and 1 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRC1_M								CRC1_L							

Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRC0_M								CRC0_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRC1_M	**The Most CRC byte for sector 1.**
23:16	CRC1_L	**The Least CRC byte for sector 1.**
15:8	CRC0_M	**The Most CRC byte for sector 0**
7:0	CRC0_L	**The Least CRC byte for sector 0.**

1100D124 NFI_CRC23 NFI CRC Data for Sector 2 and 3 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRC3_M								CRC3_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRC2_M								CRC2_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRC3_M	**The Most CRC byte for sector 1.**
23:16	CRC3_L	**The Least CRC byte for sector 1.**
15:8	CRC2_M	**The Most CRC byte for sector 0**
7:0	CRC2_L	**The Least CRC byte for sector 0.**

1100D128 NFI_CRC45 NFI CRC Data for Sector 4 and 5 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRC5_M								CRC5_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRC4_M								CRC4_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRC5_M	**The Most CRC byte for sector 1.**
23:16	CRC5_L	**The Least CRC byte for sector 1.**
15:8	CRC4_M	**The Most CRC byte for sector 0**
7:0	CRC4_L	**The Least CRC byte for sector 0.**

1100D12C NFI_CRC67 NFI CRC Data for Sector 6 and 7 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRC7_M								CRC7_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRC6_M								CRC6_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRC7_M	**The Most CRC byte for sector 1.**
23:16	CRC7_L	**The Least CRC byte for sector 1.**
15:8	CRC6_M	**The Most CRC byte for sector 0**
7:0	CRC6_L	**The Least CRC byte for sector 0.**

1100D130 NFI_CRC89 NFI CRC Data for Sector 8 and 9 Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRC9_M								CRC9_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRC8_M								CRC8_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRC9_M	**The Most CRC byte for sector 1.**
23:16	CRC9_L	**The Least CRC byte for sector 1.**
15:8	CRC8_M	**The Most CRC byte for sector 0**
7:0	CRC8_L	**The Least CRC byte for sector 0.**

1100D134 NFI_CRCAB NFI CRC Data for Sector A and B Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCB_M								CRCB_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRCA_M								CRCA_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRCB_M	**The Most CRC byte for sector 1.**
23:16	CRCB_L	**The Least CRC byte for sector 1.**
15:8	CRCA_M	**The Most CRC byte for sector 0**
7:0	CRCA_L	**The Least CRC byte for sector 0.**

1100D138 NFI_CRCCD NFI CRC Data for Sector C and D Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCD_M								CRCD_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRCC_M								CRCC_L							

Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRCD_M	The Most CRC byte for sector 1.
23:16	CRCD_L	The Least CRC byte for sector 1.
15:8	CRCC_M	The Most CRC byte for sector 0
7:0	CRCC_L	The Least CRC byte for sector 0.

1100D13C NFI_CRCEF NFI CRC Data for Sector E and F Register FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRCF_M								CRCF_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRCE_M								CRCE_L							
Type	RO								RO							
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:24	CRCF_M	The Most CRC byte for sector 1.
23:16	CRCF_L	The Least CRC byte for sector 1.
15:8	CRCE_M	The Most CRC byte for sector 0
7:0	CRCE_L	The Least CRC byte for sector 0.

1100D190 NFI_FIFODATA0 NFI FIFO Content Data 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA0	

1100D194 NFI_FIFODATA1 NFI FIFO Content Data 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:0	FIFO_DATA1

1100D198 NFI_FIFODATA2 NFI FIFO Content Data 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA2	

1100D19C NFI_FIFODATA3 NFI FIFO Content Data 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FIFO_DATA3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FIFO_DATA3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FIFO_DATA3	

1100D200 NFI_MCON NFI LCD Monitor Control Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															BMCLR	BMSTR
Type															WO	RW
Reset															0	0

Bit(s)	Name	Description
1	BMCLR	**Clear NFI-LCD bandwidth monitor register counter**
0	BMSTR	**Enable NFI-LCD bandwidth monitor** 0: disable. 1: enable.

1100D204 NFI_TOTALCNT NFI LCD Monitor Total Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NFI_TOTALCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NFI_TOTALCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_TOTALCNT	The total clock cycle count during enabling NFI-LCD bandwidth monitor

1100D208 NFI_RQCNT NFI LCD Monitor Request Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NFI_RQCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NFI_RQCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_RQCNT	The request clock cycle count during enabling NFI-LCD bandwidth monitor

1100D20C NFI_ACCNT NFI LCD Monitor Access Cycle Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	NFI_ACCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	NFI_ACCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	NFI_ACCNT	The access clock cycle count during enabling NFI-LCD bandwidth monitor

1100D220 NFI_DEBUG_CON1 NFI Debug register 0004

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BYPASS_MASTER_EN	REG_CE_HOLD	CLK_HOLD									STROBE_SEL		WBUF_EN	HWDCM_SWCON_ON	HWDCM_SWCON_EN
Type	RW	RW	RW									RW		RW	RW	RW
Reset	0	0	0									0	0	1	0	0

Bit(s)	Name	Description
15	BYPASS_MASTER_EN	**To improve performance, Bypass master async circuit** Note: just for SLC NAND, and clock@133MHz or 26MHz,otherwise this bit can not set 1 1'b1: bypass master async circuit 1'b0: not bypass master async circuit
14	REG_CE_HOLD	**this bit is used for keeping ce low if there is no data driven out during program data**

		1'b1: when program data ,ce will keep low; 1'b0: when program data,ce may release low if no data driven out.
13	CLK_HOLD	**this bit is used for keeping ONFI clock is active.** 1'b1: non-stop ONFI NAND clock during idle state 1'b0: stop ONFI clock during idle state
4:3	STROBE_SEL	**strobe select** 2'b00: sampling at the rising edge of NREB 2'b01: sampling at 1 cycle delay of the rising edge of NREB 2'b10: sampling at 2 cycles delay of the rising edge of NREB 2'b11: sampling at 3 cycles delay of the rising edge of NREB
2	WBUF_EN	
1	HWDCM_SWCON_ON	**ECC clock gating control while nfiecc is idle and nand is busy** 1'b0: ECC clock gating disable, ECC clock will not be closed while NFI is working 1'b1: ECC clock gating enable, ECC clock will be closed while ECC is idle and nand is busy
0	HWDCM_SWCON_EN	**Hard ware DCM control enable,(this bit is not used in 6571)**

1100D224 NFI_MASTER_STA NFI Master Status 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					MAS_ADDR			MAS_RD			MAS_WR			MAS_RDDLY		
Type					RO			RO			RO			RO		
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
11:9	MAS_ADDR	**MAS_is in the Address phase of AHB protocol. In this phase, Bus gots the address data from Master.** 000b: There is no MAS in the Address phase of AHB protocol. 001b: NFI is in the Address pahse of AHB protocol. 010b: Auto-Correction is in the Address pahse of AHB protocol. 100b: ECC is in the Address pahse of AHB protocol.
8:6	MAS_RD	**MAS_is in the Read DATA phase of AHB protocol. In this phase, Bus returns the read data.**
5:3	MAS_WR	**MAS_is in the Write DATA phase of AHB protocol. In this phase, Bus receives the write data.**
2:0	MAS_RDDLY	**MAS is in the Read DATA delay phase of AHB protocol. In this phase, NFI and ECC got the read back data**

1100D228 NFI_MASTER_RST pad macro soft reset 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															PAD_MACRO_RST	NFIECC_MASTER_SW_RST
Type															RW	RW
Reset															0	0

Bit(s)	Name	Description
1	PAD_MACRO_RST	1: reset

		0: de_reset
0	NFIECC_MASTER_SW_RST	1: reset 0: de_reset

1100D22C NFI_SECCUS_SIZE Enable bit for customized size for each sector 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																SECCUS_SIZE_EN
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				CUS_SEC_SIZE												
Type				RW												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
16	SECCUS_SIZE_EN	**Enable Customized sector size. Note: The spare_size will be ignored if this feature Is enabled. Autofmt and ECC function will not work under this mode.**
12:0	CUS_SEC_SIZE	**The valid size range is 1 to 8187**

1100D230 NFI_SPIADDRCNTR NFI AHB Start Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													NF_TSF_SEC			
Type													RO			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				NF_TSF_ADDR												
Type				RO												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
19:16	NF_TSF_SEC	**The register represents the start address for DMA to access EMI. These memory from the start address is used to put read data from NAND or write data to NAND in DMA mode**
12:0	NF_TSF_ADDR	

1100D234 NFI_SPIBYTELEN NFI DMA Byte Length Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													BUS_TSF_SEC			
Type													RO			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				BUS_TSF_ADDR												
Type				RO												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

19:16	BUS_TSF_SEC	**The register represents the current transfer length for DMA to access EMI.**
12:0	BUS_TSF_ADDR	

1100D238 NFI_RANDOM_CNFG Randomizer Config 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DECODED_SEED															DECODED_EN
Type	RW															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENCODED_SEED															ENCODED_EN
Type	RW															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:17	DECODED_SEED	**Randomize decoded seed**
16	DECODED_EN	**randomize decoded enable**
15:1	ENCODED_SEED	**Randomize encoded seed**
0	ENCODED_EN	**randomize encoded enable**

1100D23C NFI_EMPTY_THRESH Empty threshold setting 0000000A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ZERO_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									EMPTY_THRESH							
Type									RW							
Reset									0	0	0	0	1	0	1	0

Bit(s)	Name	Description
31:16	ZERO_CNT	**zero counter of read data in each read operation(sector /page)**
7:0	EMPTY_THRESH	**empty threshold**

1100D240 NFI_NAND_TYPE_CNFG toggle/synchronous/asynchronous interface and timing mode setting 00000004

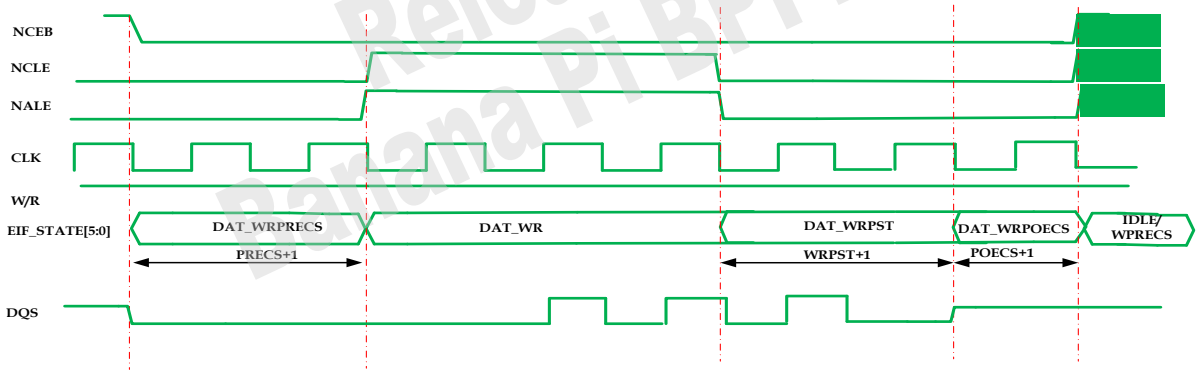
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														SLC_TYPE	INTERFACE_TYPE	
Type														RO	RW	
Reset														1	0	0

Bit(s)	Name	Description
2	SLC_TYPE	1'b0: MLC 1'b1: SLC
1:0	INTERFACE_TYPE	**interface type configuration** 2'b00: Async 2'b01: Toggle 2'b10: Sync 2'b11: Reserved

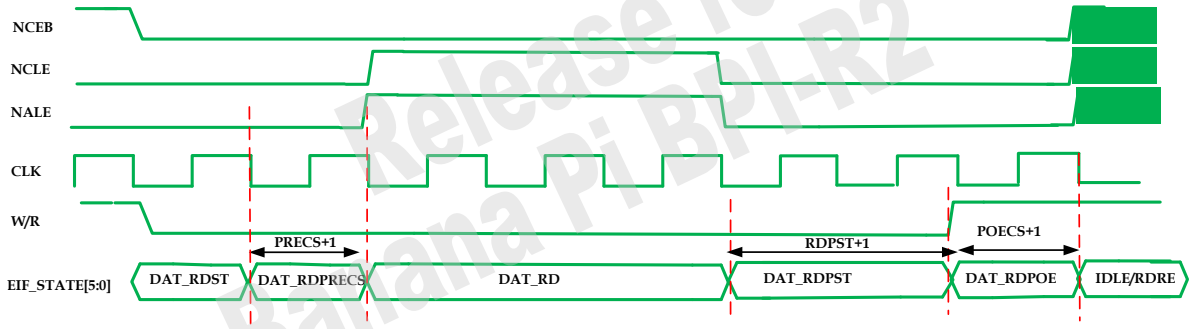
1100D244 NFI_ACCCON1 NFI access timing setting1 3F3F3F3F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			RDPRE_LATENCY								WRPRE_LATENCY					
Type			RW								RW					
Reset			1	1	1	1	1	1			1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			RDPST_LATENCY								WRPST_LATENCY					
Type			RW								RW					
Reset			1	1	1	1	1	1			1	1	1	1	1	1

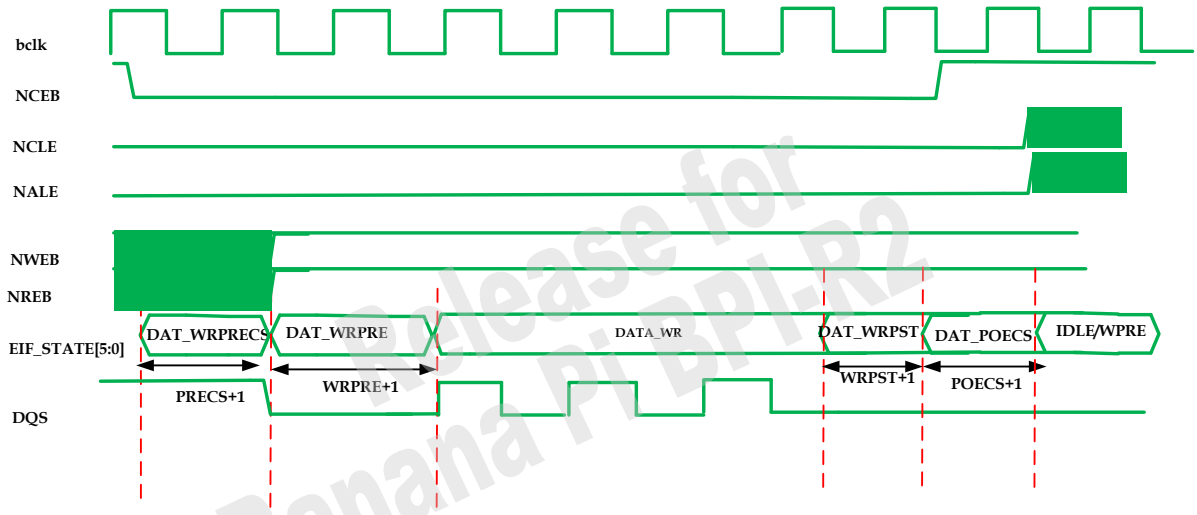
Bit(s)	Name	Description
29:24	RDPRE_LATENCY	**this field represents RDPRE state latency:RDPRE_LATENCY+1**
21:16	WRPRE_LATENCY	**this field represents WRPRE state latency:WRPRE_LATENCY+1**
13:8	RDPST_LATENCY	**this field represents RDPST state latency:RDPST_LATENCY+1**
5:0	WRPST_LATENCY	**this field represents WRPST state latency:WRPST_LATENCY+1**



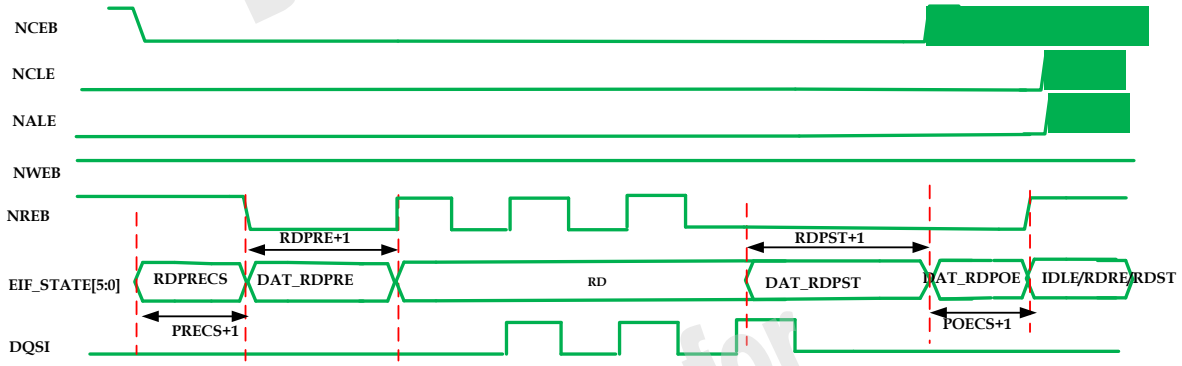
Write Data Waveform for ONFI NAND



Read Data Waveform for ONFI NAND



Write Data Waveform for Toggle NAND



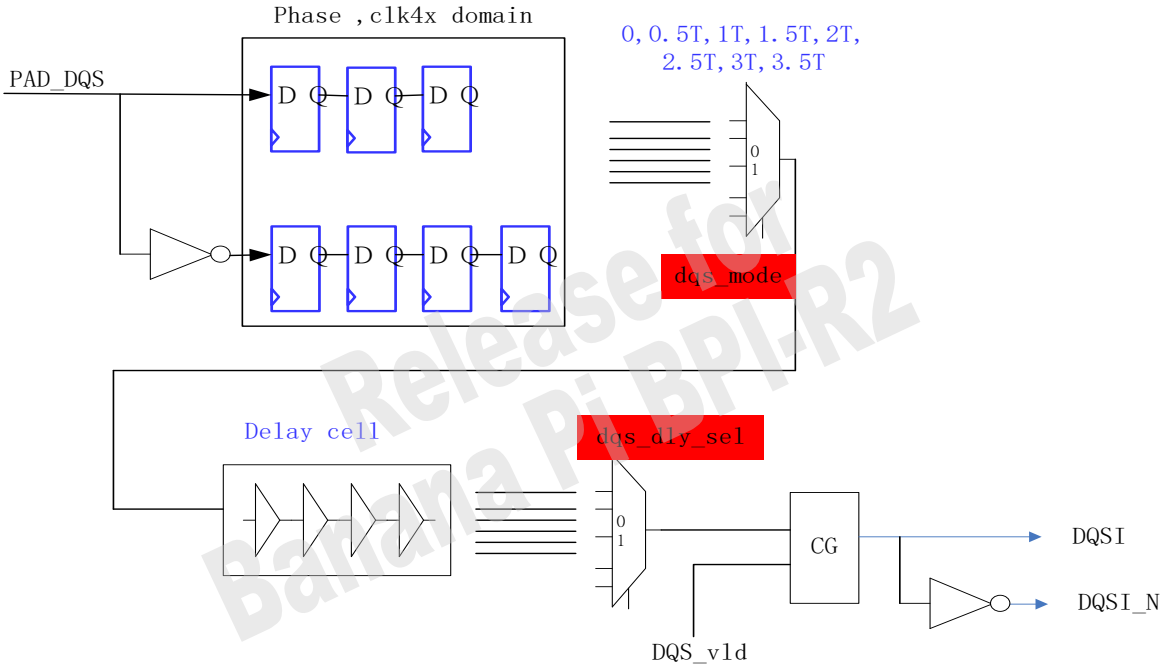
Read Data Waveform for ONFI NAND

1100D248 **NFI_DELAY_CT** **DQS and DQ delay control** **00000000**
RL

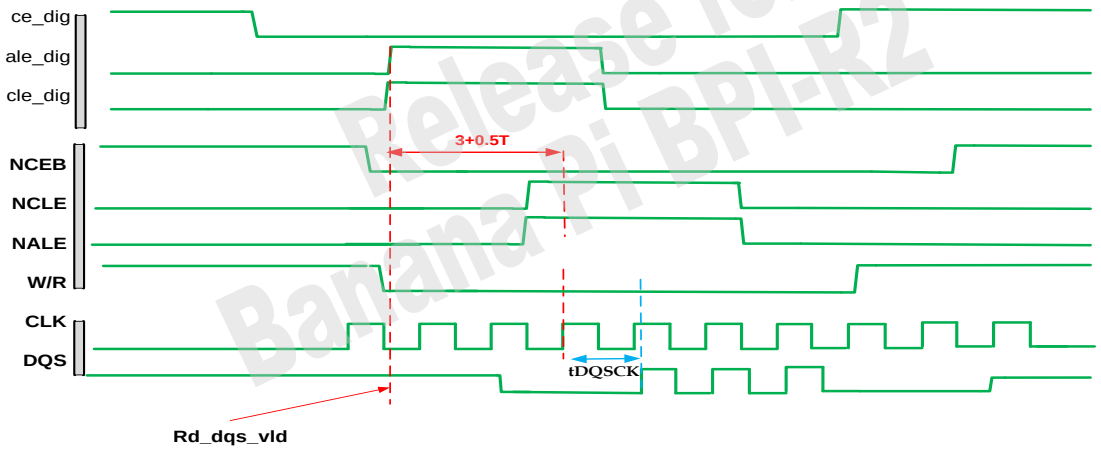
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																

Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RD_DQS_DDR_TIMING						DQS_DLY_SEL							DQS_MODE		
Type	RW						RW							RW		
Reset	0	0	0	0			0	0	0	0	0	0	0	0	0	0

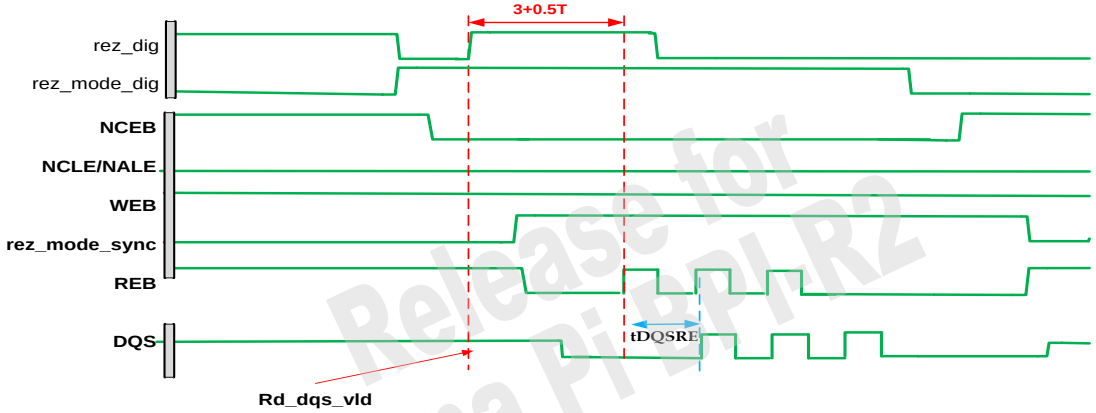
Bit(s)	Name	Description
15:12	RD_DQS_DDR_TIMING	this filed represents strobe of latch read data since rd_dqs_vld(see figure below) is valid, there are 16 types,varied from 0T to 15T,T means period of operation clock .
9:3	DQS_DLY_SEL	this filed represents delay control of DQS, there are 32 levels of delay control,and varied from 0 ns to 7.2 ns.
2:0	DQS_MODE	this filed represents delay control of DQS, there are 8 levels of delay control,and varied from 0T to 3.5T,1T means period of DQS/4.



DQS Delay Control



ONFI NAND Rd_Dqs_DDR_Timing



Toggle NAND Rd_Dqs_DDR_Timing

1100D24C NFI_RANDOM_ENSEED01_TS TOSHIBA encode seed0 seed1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EN_SEED1[16:1]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EN_SEED1[0:0]			EN_SEED0												
Type	RW			RW												
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:15	EN_SEED1	
12:0	EN_SEED0	

1100D250 NFI_RANDOM_ TOSHIBA encode seed2 00000000

ENSEED2_TS

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														EN_SEED2[18:16]		
Type														RW		
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EN_SEED2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
18:0	EN_SEED2	

1100D254 NFI_RANDOM_ENSEED3_TS TOSHIBA encode seed3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										EN_SEED3[22:16]						
Type										RW						
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EN_SEED3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
22:0	EN_SEED3	

1100D258 NFI_RANDOM_ENSEED45_TS TOSHIBA encode seed4 seed5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EN_SEED5[16:1]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EN_SEED5[0:0]			EN_SEED4												
Type	RW			RW												
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:15	EN_SEED5	
12:0	EN_SEED4	

1100D25C NFI_RANDOM_ENSEED6_TS TOSHIBA encode seed6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														EN_SEED6[18:16]		
Type														RW		
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	EN_SEED6[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
18:0	EN_SEED6	

1100D260 NFI_RANDOM_ENSEED7_TS TOSHIBA encode seed7 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										EN_SEED7[22:16]						
Type										RW						
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EN_SEED7[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
22:0	EN_SEED7	

1100D264 NFI_RANDOM_DESEED01_TS TOSHIBA encode seed0 seed1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DE_SEED1[16:1]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED1[0:0]			DE_SEED0												
Type	RW			RW												
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:15	DE_SEED1	
12:0	DE_SEED0	

1100D268 NFI_RANDOM_DESEED2_TS TOSHIBA encode seed2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														DE_SEED2[18:16]		
Type														RW		
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

18:0	DE_SEED2

1100D26C NFI_RANDOM_DESEED3_TS TOSHIBA encode seed3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										DE_SEED3[22:16]						
Type										RW						
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
22:0	DE_SEED3	

1100D270 NFI_RANDOM_DESEED45_TS TOSHIBA encode seed4 seed5 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DE_SEED5[16:1]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED5[0:0]			DE_SEED4												
Type	RW			RW												
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:15	DE_SEED5	
12:0	DE_SEED4	

1100D274 NFI_RANDOM_DESEED6_TS TOSHIBA encode seed6 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														DE_SEED6[18:16]		
Type														RW		
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED6[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
18:0	DE_SEED6	

1100D278 NFI_RANDOM_DESEED7_TS TOSHIBA encode seed7 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										DE_SEED7[22:16]						
Type										RW						
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DE_SEED7[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
22:0	DE_SEED7	

1100D500 SNF_MAC_CTL Serial nand flash mac mode control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												MAC_XIO_SEL	SF_MAC_EN	SF_TRIG	WIP_READY	WIP
Type												RW	RW	RW	RO	RO
Reset												0	0	0	0	0

Bit(s)	Name	Description
4	MAC_XIO_SEL	**MAC mode for QPI/SPI setting** 0: SPI 1: QPI
3	SF_MAC_EN	**Switches the serial flash control to update the macro. Please set up this bit before triggering the update macro (It is suggested to run MAC mode in internal sysram code due to DIRECT/MAC cannot run at the same time. Another way to switch to hardware auto switch mode by setting up MAC_MASK_OP)** 0: Disable 1: Enable (Direct read is forbidden when SF_MAC_EN = 1)
2	SF_TRIG	**Serial flash write macro trigger** 0: Disable 1: Enable (Fill command sequence I/O length before SF_TRIG)
1	WIP_READY	**WIP register status ready for access. WIP_READY exits due to asynchronous latency delay before flash responds to WIP** 0: WIP not ready for read 1: WIP ready for read (Check if WIP_READY = 1 before the next command sequence)
0	WIP	**Serial flash command write in process** 0: Flash update finished (Check if WIP = 0 before the next command sequence) 1: Not finished

1100D504 SNF_MAC_OUTL Serial nand flash mac mode output data length 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SNF_MAC_OUT_LENGTH							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	SNF_MAC_OUT_LENGTH	**Serial flash write data length**

1100D508 SNF_MAC_INL Serial nand flash mac mode input data length 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SNF_MAC_IN_LENGTH							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Name	Description
7:0	SNF_MAC_IN_LENGTH	**Serial flash read data length**

1100D50C SNF_RD_CTL1 Serial nand flash read control 1 13000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PAGE_READ_CMD								PAGE_READ_ADDRESS[23:16]							
Type	RW								RW							
Reset	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PAGE_READ_ADDRESS[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	PAGE_READ_CMD	**Page read command setting**
23:0	PAGE_READ_ADDRESS	**Page Read Address**

1100D510 SNF_RD_CTL2 Serial nand flash read control 2 0000080B

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					DATA_READ_DUMMY				DATA_READ_CMD							
Type					RW				RW							
Reset					1	0	0	0	0	0	0	0	1	0	1	1

Bit(s)	Name	Description
11:8	DATA_READ_DUMMY	**Dummy Cycle**
7:0	DATA_READ_CMD	**Data Read command**

1100D514 SNF_RD_CTL3 Serial nand flash read control 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DATA_READ_CMD_DUMMY_OUT				DATA_READ_ADDRESS											
Type	RW				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:12	DATA_READ_CMD_DUMMY_OUT	**Data read command output in dummy cycle [3:0]**
11:0	DATA_READ_ADDRESS	**Data read address setting [11:0]**

1100D518 SNF_GF_CTL1 Serial nand flash get feature control 1 0FC00101

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GF_CMD								GF_ADDR							
Type	RW								RW							
Reset	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GF_BUSY_MASK								GF_STATUS							
Type	RW								RO							
Reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:24	GF_CMD	**Get Feature command setting**
23:16	GF_ADDR	**Get Feature address setting**

15:8	GF_BUSY_MASK	**Get Feature Status busy mask bits setting**
		Ex:
		flag[7:0] = get_feature_status[7:0] & get_feature_busy [7:0];
		If (flag[7:0] == 0) done;
		Else Continue;

7:0	GF_STATUS	**Get Feature status result [7:0]**

1100D520 SNF_GF_CTL3 Serial nand flash mac mode control 000F0320

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													LOOP_LIMIT			
Type													RW			
Reset													1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	POOLING_CYCLE															
Type	RW															
Reset	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0

Bit(s)	Name	Description
19:16	LOOP_LIMIT	**Get Feature loop setting for status read and compare** NOTE: 4'b1111 for no limit setting
15:0	POOLING_CYCLE	**Polling cyce setting for standby period between issue Get Feature commands** Standby period = polling_cycle X base_time_slot Base_time_slot = spi-nand clock cycle X 128. Ex: spi-nand clock=100M, 1T=10ns, then base_time_slot = 1.28us. And the polling cycle value should be set to meet the Standby_period ~= spec time.

1100D524 SNF_PG_CTL1 Serial nand flash program control1 00100206

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									PG_EXE_CMD							
Type									RW							
Reset									0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PG_LOAD_CMD								WRITE_EN_CMD							
Type	RW								RW							
Reset	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0

Bit(s)	Name	Description
23:16	PG_EXE_CMD	**Program execute command [7:0]**
15:8	PG_LOAD_CMD	**Program load command [7:0]**
7:0	WRITE_EN_CMD	**Write Enable command [7:0]**

1100D528 SNF_PG_CTL2 Serial nand flash program control2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PG_LOAD_CMD_DUMMY_OUT								PG_LOAD_ADDR							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:8	PG_LOAD_CMD_DUMMY_OUT	**Program load command output in dummy cycle [3:0]**
7:0	PG_LOAD_ADDR	**Program load address [7:0]**

1100D52C SNF_PG_CTL3 Serial nand flash program control3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									PG_EXE_ADDR[23:16]							
Type									RW							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PG_EXE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
23:0	PG_EXE_ADDR	**Program execute address [23:0]**

1100D530 SNF_ER_CTL Serial nand flash erase control 0000D800

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERASE_CMD															AUTO

																_ERASE_TRIGGER
Type	RW															RW
Reset	1	1	0	1	1	0	0	0								0

Bit(s)	Name	Description
15:8	ERASE_CMD	**Erase command setting [7:0]**
0	AUTO_ERASE_TRIGGER	**Auto Ease trigger bit**

1100D534 SNF_ER_CTL2 Serial nand flash erase control 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ERASE_ADDR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERASE_ADDR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ERASE_ADDR	**Erase address setting [23:0]**

1100D538 SNF_MISC_CTL Serial nand flash MISC control 0400000A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			MACRO_RST_EN	SW_RST	SFIO_EN_SEL	FIFO_RD_LTC		_4FIFO_EN	FBCLK_SEL	SMPCK_INV	CLK_INV	PG_LOAD_X4_EN		DATA_READ_MODE		
Type			RW	RW	RW	RW		RW	RW	RW	RW	RW		RW		
Reset			0	0	0	1	0	0	0	0	0	0		0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		SF2CS_DUAL_EN	SF2CS_SEL	SF2CS_EN			LATCH_LAT		PG_LOAD_CUSTOM_EN	DATARD_CUSTOM_EN		CS_DESELECT_CYC				
Type		RW	RW	RW			RW		RW	RW		RW				
Reset		0	0	0			0	0	0	0		0	1	0	1	0

Bit(s)	Name	Description
29	MACRO_RST_EN	**MACRO reset fifo enable**
28	SW_RST	**SW reset controller: 1 for reset on, 0 for reset off.**
27	SFIO_EN_SEL	**IO PAD Enable signal align with clock positive edge select**
26:25	FIFO_RD_LTC	**FIFO read latency**
24	_4FIFO_EN	**4Tap FIFO read enable**
23	FBCLK_SEL	**Feedback clock select**
22	SMPCK_INV	**Inner sample clock inverter select**
21	CLK_INV	**Output serial clock inverter select**
20	PG_LOAD_X4_EN	**Program load data x4 mode enable.**
18:16	DATA_READ_MODE	**Data Read mode select. Bit[1:0] for data mode, Bit[2] for IO mode.** 000: X1 data mode

		001: X2 data mode 010: X4 data mode 011: Reserved 100: Reserved 101: Dual IO mode 110: Quad IO mode 111: Reserved
14	SF2CS_DUAL_EN	**Both CS1/CS2 enable/select for first/second SPI-nand device.**
13	SF2CS_SEL	**SF 2CS select for first/second SPI-nand device.**
12	SF2CS_EN	**SF 2CS enable for second SPI-nand device.**
9:8	LATCH_LAT	**Data read latch latency**
7	PG_LOAD_CUSTOM_EN	**Program load custom mode enable**
6	DATARD_CUSTOM_EN	**Data read custom mode enable**
4:0	CS_DESELECT_CYC	**CS deselect cycle setting**

1100D53C SNF_MISC_CTL2 Serial nand flash MISC control 2 02100210

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					PROGRAM_LOAD_BYTE_NUM											
Type					RW											
Reset					0	0	1	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					READ_DATA_BYTE_NUM											
Type					RW											
Reset					0	0	1	0	0	0	0	1	0	0	0	0

Bit(s)	Name	Description

27:16	PROGRAM_LOAD_BYTE_NUM	**NOTE: Setting value should be sync with NFI Read/Write byte length.**
		if (NFI_SECCUS_SIZE.CUS_SEC_SIZE.SECCUS_SIZE_EN==0) {
		(512 + PAGEFMT.spare_size) * NFI_CON.sec_num
		== Transfer Data Length
		== SNF_MISC_CTL2.PROGRAM_LOAD_BYTE_NUM
		== SNF_MISC_CTL2.READ_DATA_BYTE_NUM
		} else {
		(NFI_SECCUS_SIZE.CUS_SEC_SIZE) * NFI_CON.sec_num
		== Transfer Data Length
		== SNF_MISC_CTL2.PROGRAM_LOAD_BYTE_NUM
		== SNF_MISC_CTL2.READ_DATA_BYTE_NUM

11:0	READ_DATA_BYTE_NUM	

1100D540 SNF_DLY_CTL1 Serial nand flash delay control setting 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			SFIO3_OUT_DLY								SFIO2_OUT_DLY					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			SFIO1_OUT_DLY										SFIO0_OUT_DLY			
Type			RW										RW			
Reset			0	0	0	0	0	0					0	0	0	0

Bit(s)	Name	Description
29:24	SFIO3_OUT_DLY	**Serial flash SFIO3 pin IO output delay setting**
21:16	SFIO2_OUT_DLY	**Serial flash SFIO2 pin IO output delay setting**

13:8	SFIO1_OUT_DLY	**Serial flash SFIO1 pin IO output delay setting**
3:0	SFIO0_OUT_DLY	**Serial flash SFIO0 pin IO output delay setting**

1100D544 SNF_DLY_CTL2 Serial nand flash delay control setting 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			SFIO3_IN_DLY								SFIO2_IN_DLY					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			SFIO1_IN_DLY								SFIO0_IN_DLY					
Type			RW								RW					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Bit(s)	Name	Description
29:24	SFIO3_IN_DLY	**Serial flash SFIO3 pin IO input delay setting**
21:16	SFIO2_IN_DLY	**Serial flash SFIO2 pin IO input delay setting**
13:8	SFIO1_IN_DLY	**Serial flash SFIO1 pin IO input delay setting**
5:0	SFIO0_IN_DLY	**Serial flash SFIO0 pin IO input delay setting**

1100D548 SNF_DLY_CTL3 Serial nand flash control setting 3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			SFIFO_WR_EN_DLY_SEL										SFCS_DLY			
Type			RW										RW			
Reset			0	0	0	0	0	0					0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					SFCK_OUT_DLY						SFCK_SAM_DLY					
Type					RW						RW					
Reset					0	0	0	0			0	0	0	0	0	0

Bit(s)	Name	Description
29:24	SFIFO_WR_EN_DLY_SEL	**Serial flash FIFO write enable delay select setting**
19:16	SFCS_DLY	**Serial flash SFCS pin IO output delay setting**
11:8	SFCK_OUT_DLY	**Serial flash CK pin IO output delay setting**
5:0	SFCK_SAM_DLY	**Serial flash sample clock delay setting**

1100D54C SNF_DLY_CTL4 Serial nand flash delay control setting 4 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													SFCS2_DLY			
Type													RW			
Reset													0	0	0	0

Bit(s)	Name	Description
3:0	SFCS2_DLY	**Serial flash SFCS2 pin IO output delay setting**

1100D550 SNF_STA_CTL1 **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			GF_LOOP_TIMEOUT	CUS_PG_DONE	CUS_READ_DONE	AUTO_PG_DONE	AUTO_READ_DONE	AUTO_BLRER_DONE		DATARD_STATE			PGREAD_STATE			PGEXE_STATE[2:2]
Type			RO	W1C	W1C	W1C	W1C	W1C		RO			RO			RO
Reset			0	0	0	0	0	0		0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PGEXE_STATE[1:0]		PGLOAD_STATE			GF_STATE			BLKER_STATE			WREN_STATE	SPI_STATE			
Type	RO		RO			RO			RO			RO	RO			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
29	GF_LOOP_TIMEOUT	**Get Feature read status time out flag**
28	CUS_PG_DONE	**Custom program mode Clear setting and Status** Status read(R): Custom program mode done flag. Clear(W): Set 1 then Set 0 to clear this flag.
27	CUS_READ_DONE	**Custom read mode Clear setting and Status** Status read(R): Custom read mode done flag. Clear(W): Set 1 then Set 0 to clear this flag.
26	AUTO_PG_DONE	**Auto program mode Clear setting and Status** Status read(R): Auto program mode done flag. Clear(W): Set 1 then Set 0 to clear this flag.
25	AUTO_READ_DONE	**Auto read mode Clear setting and Status** Status read(R): Auto read mode done flag. Clear(W): Set 1 then Set 0 to clear this flag.
24	AUTO_BLRER_DONE	**Auto block erase mode Clear setting and Status** Status read(R): Auto block erase mode done flag. Clear(W): Set 1 then Set 0 to clear this flag.
22:20	DATARD_STATE	**State machine of custom read** 0: IDLE 1: CMD 2: ADDR1 3: ADDR2 4: DUMMY 5: DATA 6: BUFOUT 7: WAIT
19:17	PGREAD_STATE	**State machine of auto read** 0: IDLE 1: CMD 2: ADDR1 3: ADDR2 4: ADDR3
16:14	PGEXE_STATE	**State machine of program execution** 0: IDLE 1: CMD 2: ADDR1 3: ADDR2 4: ADDR3
13:11	PGLOAD_STATE	**State machine of program load** 0: IDLE 1: CMD 2: ADDR1 3: ADDR2 4: BUFIN 5: DATA

10:8	GF_STATE	**State machine of get feature** 0: IDLE 1: GF_CMD 2: GF_ADDR 3: GF_DATA 4: GF_CMP 5: GF_WAIT
7:5	BLKER_STATE	**State machine of auto block erase** 0: IDLE 1: CMD 2: ADDR1 3: ADDR2 4: ADDR3
4	WREN_STATE	**State machine of write enable** 0: IDLE 1: CMD
3:0	SPI_STATE	**State machine SPI main Controller** 0: IDLE 1: Write Enable 2: Block Erase 3: Get Feature 4: Program Load 5: Program Execution 6: Page Read 7: Custom Data Read

1100D554 SNF_STA_CTL2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name				DATARD_BYTE_CNT												
Type				RO												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				PGLOAD_BTTE_CNT												
Type				RO												
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
28:16	DATARD_BYTE_CNT	**Data Read Transfer byte count**
12:0	PGLOAD_BTTE_CNT	**Program Load Transfer byte count**

1100D558 SNF_STA_CTL3 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										GF_BASE_CNT						
Type										RO						
Reset										0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GF_WAIT_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
22:16	GF_BASE_CNT	**Get feature base counter**
15:0	GF_WAIT_CNT	**Get feature wait counter**

1100D55C SNF_SNF_CNFG SPI/Parallel NAND Selection 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SPI_MODE
Type																RW
Reset																0

Bit(s)	Name	Description
0	SPI_MODE	**Switch for Parallel NAND or Serial NAND** 0: NFI 1: SPI NAND

1100D560 SNF_DEBUG_SEL DEBUG MUX Selection 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEBUG_SEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
15:0	DEBUG_SEL	

14.5 NFIECC Register Definition

Module name: NFIECC Base address: (+1100E000h)

Address	Name	Width	Register Function
1100E000	**NFIECC_ENCCON**	16	**NFIECC Encoder Control Register** This register is for Encoder control.
1100E004	**NFIECC_ENCCNFG**	32	**NFIECC Configure Register** This register is for NFIECC encoder configuration.
1100E008	**NFIECC_ENCDIADDR**	32	**NFIECC Encoder DI Memory Address Register** The register indicates the data start address of input data to the Encoder AHB mode.
1100E00C	**NFIECC_ENCIDLE**	32	**NFIECC Encoder Idle Status Register** This register is for NFIECC Encoder idle status.
1100E010	**NFIECC_ENCPAR0**	32	**NFIECC Parity0 Register** The register indicates the highest order of parity bits
1100E014	**NFIECC_ENCPAR1**	32	**NFIECC Parity1 Register** The register indicates the parity bits
1100E018	**NFIECC_ENCPAR2**	32	**NFIECC Parity2 Register** The register indicates the parity bits

1100E01C	**NFIECC_ENCPAR3**	32	**NFIECC Parity3 Register** The register indicates the parity bits
1100E020	**NFIECC_ENCPAR4**	32	**NFIECC Parity4 Register** The register indicates the parity bits
1100E024	**NFIECC_ENCPAR5**	32	**NFIECC Parity5 Register** The register indicates the parity bits
1100E028	**NFIECC_ENCPAR6**	32	**NFIECC Parity6 Register** The register indicates the parity bits
1100E02C	**NFIECC_ENCPAR7**	32	**NFIECC Parity7 Register** The register indicates the parity bits
1100E030	**NFIECC_ENCPAR8**	32	**NFIECC Parity8 Register** The register indicates the parity bits
1100E034	**NFIECC_ENCPAR9**	32	**NFIECC Parity9 Register** The register indicates the parity bits
1100E038	**NFIECC_ENCPAR10**	32	**NFIECC Parity10 Register** The register indicates the parity bits
1100E03C	**NFIECC_ENCPAR11**	32	**NFIECC Parity11 Register** The register indicates the parity bits
1100E040	**NFIECC_ENCPAR12**	32	**NFIECC Parity12 Register** The register indicates the parity bits
1100E044	**NFIECC_ENCPAR13**	32	**NFIECC Parity13 Register** The register indicates the parity bits
1100E048	**NFIECC_ENCPAR14**	32	**NFIECC Parity14 Register** The register indicates the parity bits
1100E04C	**NFIECC_ENCPAR15**	32	**NFIECC Parity15 Register** The register indicates the parity bits
1100E050	**NFIECC_ENCPAR16**	32	**NFIECC Parity16 Register** The register indicates the parity bits
1100E054	**NFIECC_ENCPAR17**	32	**NFIECC Parity17 Register** The register indicates the parity bits
1100E058	**NFIECC_ENCPAR18**	32	**NFIECC Parity18 Register** The register indicates the parity bits
1100E05C	**NFIECC_ENCPAR19**	32	**NFIECC Parity19 Register** The register indicates the parity bits
1100E060	**NFIECC_ENCPAR20**	32	**NFIECC Parity20 Register** The register indicates the parity bits
1100E064	**NFIECC_ENCPAR21**	32	**NFIECC Parity21 Register** The register indicates the parity bits
1100E068	**NFIECC_ENCPAR22**	32	**NFIECC Parity22 Register** The register indicates the parity bits
1100E06C	**NFIECC_ENCPAR23**	32	**NFIECC Parity23 Register** The register indicates the parity bits
1100E070	**NFIECC_ENCPAR24**	32	**NFIECC Parity24 Register** The register indicates the parity bits
1100E074	**NFIECC_ENCPAR25**	32	**NFIECC Parity25 Register** The register indicates the parity bits
1100E078	**NFIECC_ENCPAR26**	32	**NFIECC Parity26 Register** The register indicates the parity bits
1100E07C	**NFIECC_ENCSTA**	32	**NFIECC Encoder Status Register** This register is for NFIECC Encoder status for SW polling
1100E080	**NFIECC_ENCIRQEN**	32	**NFIECC Encoder IRQ enable Register** This register is for software programmer to enable NFIECC IRQ signals

			(ignore in NFI mode)
1100E084	**NFIECC_ENCIRQSTA**	32	**NFIECC Encoder IRQ status Register** This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)
1100E090	**NFIECC_PIO_DIRDY**	32	**NFIECC PIO Data Ready Register** This register indicates the data is ready for input
1100E094	**NFIECC_PIO_DI**	32	**NFIECC PIO Data Register** The register indicates PIO mode data input by MCU
1100E100	**NFIECC_DECCON**	32	**NFIECC Decoder Control Register** This register is for Decoder control.
1100E104	**NFIECC_DECCNFG**	32	**NFIECC Decoder Configure Register** This register is for NFIECC configuration.
1100E108	**NFIECC_DECDIADDR**	32	**NFIECC Decoder DI Memory Address Register** The register indicates the data start address of input data to the Decoder AHB mode.
1100E10C	**NFIECC_DECIDLE**	16	**NFIECC Decoder Idle Status Register** This register indicates the Decoder Idle status.
1100E110	**NFIECC_DECFER**	16	**NFIECC Decoder Found Error Status Register** This register is for NFIECC Decoder status.
1100E114	**NFIECC_DECENUM0**	32	**NFIECC Decode Error Number Register** The register indicates the error number of the coded block.
1100E118	**NFIECC_DECENUM1**	32	**NFIECC Decode Error Number Register** The register indicates the error number of the coded block.
1100E11C	**NFIECC_DECENUM2**	32	**NFIECC Decode Error Number Register** The register indicates the error number of the coded block.
1100E120	**NFIECC_DECENUM3**	32	**NFIECC Decode Error Number Register** The register indicates the error number of the coded block.
1100E124	**NFIECC_DECDONE**	16	**NFIECC Decoder Error Status Register** This register is for NFIECC Decoder done status.
1100E128	**NFIECC_DECEL0**	32	**NFIECC Decoder Error location0 Register** The register indicates the error location of the decoding result
1100E12C	**NFIECC_DECEL1**	32	**NFIECC Decoder Error location1 Register** The register indicates the error location of the decoding result
1100E130	**NFIECC_DECEL2**	32	**NFIECC Decoder Error location2 Register** The register indicates the error location of the decoding result
1100E134	**NFIECC_DECEL3**	32	**NFIECC Decoder Error location3 Register** The register indicates the error location of the decoding result
1100E138	**NFIECC_DECEL4**	32	**NFIECC Decoder Error location4 Register** The register indicates the error location of the decoding result
1100E13C	**NFIECC_DECEL5**	32	**NFIECC Decoder Error location5 Register** The register indicates the error location of the decoding result
1100E140	**NFIECC_DECEL6**	32	**NFIECC Decoder Error location6 Register** The register indicates the error location of the decoding result
1100E144	**NFIECC_DECEL7**	32	**NFIECC Decoder Error location7 Register** The register indicates the error location of the decoding result
1100E148	**NFIECC_DECEL8**	32	**NFIECC Decoder Error location8 Register** The register indicates the error location of the decoding result
1100E14C	**NFIECC_DECEL9**	32	**NFIECC Decoder Error location9 Register** The register indicates the error location of the decoding result
1100E150	**NFIECC_DECEL10**	32	**NFIECC Decoder Error location10 Register** The register indicates the error location of the decoding result
1100E154	**NFIECC_DECEL11**	32	**NFIECC Decoder Error location11 Register**

			The register indicates the error location of the decoding result
1100E158	**NFIECC_DECEL12**	32	**NFIECC Decoder Error location12 Register** The register indicates the error location of the decoding result
1100E15C	**NFIECC_DECEL13**	32	**NFIECC Decoder Error location13 Register** The register indicates the error location of the decoding result
1100E160	**NFIECC_DECEL14**	32	**NFIECC Decoder Error location14 Register** The register indicates the error location of the decoding result
1100E164	**NFIECC_DECEL15**	32	**NFIECC Decoder Error location15 Register** The register indicates the error location of the decoding result
1100E168	**NFIECC_DECEL16**	32	**NFIECC Decoder Error location16 Register** The register indicates the error location of the decoding result
1100E16C	**NFIECC_DECEL17**	32	**NFIECC Decoder Error location17 Register** The register indicates the error location of the decoding result
1100E170	**NFIECC_DECEL18**	32	**NFIECC Decoder Error location18 Register** The register indicates the error location of the decoding result
1100E174	**NFIECC_DECEL19**	32	**NFIECC Decoder Error location19 Register** The register indicates the error location of the decoding result
1100E178	**NFIECC_DECEL20**	32	**NFIECC Decoder Error location20 Register** The register indicates the error location of the decoding result
1100E17C	**NFIECC_DECEL21**	32	**NFIECC Decoder Error location21 Register** The register indicates the error location of the decoding result
1100E180	**NFIECC_DECEL22**	32	**NFIECC Decoder Error location22 Register** The register indicates the error location of the decoding result
1100E184	**NFIECC_DECEL23**	32	**NFIECC Decoder Error location23 Register** The register indicates the error location of the decoding result
1100E188	**NFIECC_DECEL24**	32	**NFIECC Decoder Error location24 Register** The register indicates the error location of the decoding result
1100E18C	**NFIECC_DECEL25**	32	**NFIECC Decoder Error location25 Register** The register indicates the error location of the decoding result
1100E190	**NFIECC_DECEL26**	32	**NFIECC Decoder Error location26 Register** The register indicates the error location of the decoding result
1100E194	**NFIECC_DECEL27**	32	**NFIECC Decoder Error location27 Register** The register indicates the error location of the decoding result
1100E198	**NFIECC_DECEL28**	32	**NFIECC Decoder Error location28 Register** The register indicates the error location of the decoding result
1100E19C	**NFIECC_DECEL29**	32	**NFIECC Decoder Error location29 Register** The register indicates the error location of the decoding result
1100E200	**NFIECC_DECIRQEN**	16	**NFIECC Decoder IRQ enable Register** This register is for software programmer to enable NFIECC IRQ signals (ignore in NFI mode)
1100E204	**NFIECC_DECIRQSTA**	16	**NFIECC_DECIRQSTA** This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)
1100E208	**NFIECC_DECFSM**	32	**NFIECC Decoder FSM** The register indicates the finite state machine status of decoder.
1100E20C	**NFIECC_BYPASS**	32	**NFIECC BYPASS Async circuit** The register used to bypass the APB async design in NFIECC

1100E000 **NFIECC_ENCCON** **NFIECC Encoder Control Register** **0000**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ENC_EN
Type																RW
Reset																0

Overview This register is for Encoder control.

Bit(s)	Name	Description
0	ENC_EN	**indicates the enable in NFI mode and start to work in AHB mode. In AHB mode, parity bits is remained in the PAR0~PAR4 register field until the ENC_EN is deasserted to 0.** 0: means disable the Encode block. 1: means enable the Encode block. In AHB mode, the Encoder starts to fetch data when the register changes from 0 to 1. In NFI mode, the register enables the Encode block, and then the Encoder module waits start signal and data from NFI.

1100E004 NFIECC_ENCCNFG NFIECC Configure Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ENC_MS													
Type			RW													
Reset			0	0	0	0	0	0	0	0	0	0	0			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								ENC_BURST_EN		ENC_MODE		ENC_TNUM				
Type								RW		RW		RW				
Reset								0		0	0	0	0	0	0	0

Overview This register is for NFIECC encoder configuration.

Bit(s)	Name	Description
29:19	ENC_MS	**indicates the total bytes size of message block including main data and control(FDM) data in the NFI mode.The message block can only be in byte unit.** The spare_ECC_num parameter in old version has been merged into the message block_size parameter. If the block_size is equal to zero, the NFIECC do nothing. The acceptable coded block size, which includes data and parity bits size, is 1~16383bits. Different ENC_TNUM results in different parity bits, and also results in different maximum message block size.
8	ENC_BURST_EN	**indicates the burst enalbe.** 1'b0: means DMA mode uses single read. 1'b1: means DMA mode uses burst read.
6:5	ENC_MODE	**indicates the data source from access through AHB bus or from NFI.** 2'b00: means source data from access through Bus. (DMA mode) 2'b01: means source data from NFI module. (NFI mode) 2'b10: means source data is written by MCU. (PIO mode) 2'b11: reserved mode.
4:0	ENC_TNUM	**indicates the correct capability in one block size.** 5'd0: means the NFIECC is capable of correct 4 bits in one block size. 5'd1: means the NFIECC is capable of correct 6 bits in one block size. 5'd2: means the NFIECC is capable of correct 8 bits in one block size. 5'd3: means the NFIECC is capable of correct 10 bits in one block size. 5'd4: means the NFIECC is capable of correct 12 bits in one block size. 5'd5: means the NFIECC is capable of correct 14 bits in one block size. 5'd6: means the NFIECC is capable of correct 16 bits in one block size. 5'd7: means the NFIECC is capable of correct 18 bits in one block size.

5'd8: means the NFIECC is capable of correct 20 bits in one block size.
5'd9: means the NFIECC is capable of correct 22 bits in one block size.
5'd10: means the NFIECC is capable of correct 24 bits in one block size.
5'd11: means the NFIECC is capable of correct 28 bits in one block size.
5'd12: means the NFIECC is capable of correct 32 bits in one block size.
5'd13: means the NFIECC is capable of correct 36 bits in one block size.
5'd14: means the NFIECC is capable of correct 40 bits in one block size.
5'd15: means the NFIECC is capable of correct 44 bits in one block size.
5'd16: means the NFIECC is capable of correct 48 bits in one block size.
5'd17: means the NFIECC is capable of correct 52 bits in one block size.
5'd18: means the NFIECC is capable of correct 56 bits in one block size.
5'd19: means the NFIECC is capable of correct 60 bits in one block size.

1100E008 NFIECC_ENCDIADDR NFIECC Encoder DI Memory Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_DIADDR[29:14]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_DIADDR[13:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Overview The register indicates the data start address of input data to the Encoder AHB mode.

Bit(s)	Name	Description
31:2	ENC_DIADDR	**indicates the memory address of input data to Encoder block in AHB mode. (4-Byte align)**

1100E00C NFIECC_ENCIDLE NFIECC Encoder Idle Status Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ENC_IDLE
Type																RO
Reset																1

Overview This register is for NFIECC Encoder idle status.

Bit(s)	Name	Description
0	ENC_IDLE	**indicates the Encode block in idle state and ready for new message block.** 0: means the Encode block is under working. 1: means the Encode block is in Idle state and available for new message block.

1100E010 NFIECC_ENCPAR0 NFIECC Parity0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR0[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the highest order of parity bits

Bit(s)	Name	Description
31:0	ENC_PAR0	**indicates the highest order of output parity bits and the bit 0 is the highest order of parity bit. The PAR0~PAR26 register is remain the last message block parity bits until ENC_EN is deasserted. The parity bits should append after main data by order of {PAR0[31:0], PAR1[31:0], PAR2[31:0], PAR3[31:0], PAR4[31:0],...,PAR26[7:0]}, The redundant bit of parity bit will be padded by 1.**

1100E014 NFIECC_ENCPAR1 NFIECC Parity1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR1	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E018 NFIECC_ENCPAR2 NFIECC Parity2 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR2[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR2[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR2	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E01C NFIECC_ENCPAR3 NFIECC Parity3 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR3[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR3[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR3	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E020 NFIECC_ENCPAR4 NFIECC Parity4 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR4[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR4[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR4	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E024 NFIECC_ENCPAR5 NFIECC Parity5 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR5[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR5[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR5	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E028 NFIECC_ENCPAR6 NFIECC Parity6 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR6[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR6[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR6	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E02C NFIECC_ENCPAR7 NFIECC Parity7 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR7[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR7[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR7	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E030 NFIECC_ENCPAR8 NFIECC Parity8 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR8[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR8[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR8	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E034 NFIECC_ENCPAR9 NFIECC Parity9 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR9[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR9[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR9	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E038 NFIECC_ENCPAR10 NFIECC Parity10 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR10[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR10[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR10	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E03C NFIECC_ENCPAR11 NFIECC Parity11 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR11[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR11[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR11	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E040 NFIECC_ENCPAR12 NFIECC Parity12 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR12[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR12[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR12	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E044 NFIECC_ENCPAR13 NFIECC Parity13 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR13[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR13[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR13	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E048 NFIECC_ENCPAR14 NFIECC Parity14 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR14[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR14[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR14	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E04C NFIECC_ENCPAR15 NFIECC Parity15 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR15[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR15[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR15	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E050 NFIECC_ENCPAR16 NFIECC Parity16 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR16[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR16[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR16	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E054 NFIECC_ENCPAR17 NFIECC Parity17 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR17[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR17[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR17	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E058 NFIECC_ENCPAR18 NFIECC Parity18 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR18[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR18[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR18	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E05C NFIECC_ENCPAR19 NFIECC Parity19 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR19[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR19[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR19	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E060 NFIECC_ENCPAR20 NFIECC Parity20 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR20[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR20[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR20	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E064 NFIECC_ENCPAR21 NFIECC Parity21 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR21[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR21[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR21	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E068 NFIECC_ENCPAR22 NFIECC Parity22 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR22[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR22[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR22	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E06C NFIECC_ENCPAR23 NFIECC Parity23 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR23[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR23[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR23	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E070 NFIECC_ENCPAR24 NFIECC Parity24 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR24[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR24[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR24	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E074 NFIECC_ENCPAR25 NFIECC Parity25 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR25[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR25[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR25	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E078 NFIECC_ENCPAR26 NFIECC Parity26 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENC_PAR26[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENC_PAR26[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the parity bits

Bit(s)	Name	Description
31:0	ENC_PAR26	**indicates the parity bits and the bit 0 is the highest order of parity bit.**

1100E07C NFIECC_ENCSTA NFIECC Encoder Status Register 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			COUNT_MS													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0			
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														ENC_FSM		
Type														RO		
Reset														0	0	1

Overview This register is for NFIECC Encoder status for SW polling

Bit(s)	Name	Description
29:19	COUNT_MS	**indicates the remaining un-processing message bytes.**
2:0	ENC_FSM	**3'd1: IDLE** 3'd2: DATA 3'd4: DONE

1100E080 NFIECC_ENCIRQEN NFIECC Encoder IRQ enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ENC_IRQEN
Type																RW
Reset																0

Overview This register is for software programmer to enable NFIECC IRQ signals (ignore in NFI mode)

Bit(s)	Name	Description
0	ENC_IRQEN	**Encoder IRQ mask: triggered when Encoder operation is completed.** 0: Disable 1: Enable

1100E084 NFIECC_ENCIRQSTA NFIECC Encoder IRQ status Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																ENC_IRQSTA
Type																RC
Reset																0

Overview This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)

Bit(s)	Name	Description
0	ENC_IRQSTA	**indicates interrupt status for Encoder processing.** 0: No interrupt is generated. 1: An interrupt is pending and waiting for service. Active when Encoder processing is done.

1100E090 NFIECC_PIO_DIRDY NFIECC PIO Data Ready Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																PIO_DI_RDY
Type																RO
Reset																0

Overview This register indicates the data is ready for input

Bit(s)	Name	Description
0	PIO_DI_RDY	**indicates the PIO mode (Encoder/Decoder) is ready for input data.** 0: ECC is busy. During busy state, NFIECC_PIO_DI should not be over-write. 1: ECC is ready for input data. In PIO mode, write next PIO_DI when pio_di_rdy is equal to 1.

1100E094 NFIECC_PIO_DI NFIECC PIO Data Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PIO_DI[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PIO_DI[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates PIO mode data input by MCU

Bit(s)	Name	Description
31:0	PIO_DI	**indicates the PIO mode data input.**

1100E100 NFIECC_DECCON NFIECC Decoder Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DEC_EN
Type																RW
Reset																0

Overview This register is for Decoder control.

Bit(s)	Name	Description
0	DEC_EN	**indicates the enable in NFI mode and start to work in AHB mode. In AHB mode, the decode-status FER and error number registers and error location registers will be reset to 0 when DEC_EN is deasserted.** 0: means disable the Decode block. 1: means enable the Decode block. In AHB mode, the Decoder starts to fetch data when the register changes from 0 to 1. In NFI mode, the register enables the Decode block, and then the Decoder module waits start signal and data from NFI.

1100E104 NFIECC_DECCNFG NFIECC Decoder Configure Register 00003000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEC_EMPTY_EN		DEC_CS													
Type	RW		RW													
Reset	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_CON					DEC_BURST_EN		DEC_MODE		DEC_TNUM				
Type			RW					RW		RW		RW				
Reset			1	1				0		0	0	0	0	0	0	0

Overview This register is for NFIECC configuration.

Bit(s)	Name	Description
31	DEC_EMPTY_EN	**indicates the Decoder automatically detects “all equal to 1” source data and by pass the auto-correction block . (ignore in AHB_mode)** 0: means disenable the detection of all equal to 1 data. 1: means enable the detection of all equal to 1 data. Note: This bit can reduce the time for decoder, and decoder will regard this kind of data as uncorrectable, because in MLC NAND flash, zeros are allowed in empty page.
29:16	DEC_CS	**indicates the total bit size of coded block including protected data and parity bits. The acceptable coded block size is 1~8191bits. If the coded block size is equal to zero, the decoder does nothing. The detail figure shows in Figure.**
13:12	DEC_CON	**indicates the bypass configuration in decoding processor.** 0: is reserved 1: means only active syndrome calculator for error detecting purpose. ECC reports DONE and FER status after syndrome calculator is done. 2: means error-correction module is bypassed for being aware of error location purpose. ECC reports DONE, FER, EL and ERRNUM status after Chien search is done. 3: means the ECC processor decoded data and auto-correction error data. The data address is signaled by DEC_DIADDR register in AHB mode and NFI_DIADDR in NFI mode. ECC reports DONE, FER, EL and ERRNUM status after error-correction is done.
8	DEC_BURST_EN	**indicates the burst enalbe.** 1'b0: means DMA mode uses single read. 1'b1: means DMA mode uses burst read.
6:5	DEC_MODE	**indicates the data source from access AHB bus ,MCU or from NFI.**

		2'b00: means source data from access through Bus. (DMA mode) 2'b01: means source data from NFI module. (NFI mode) 2'b10: means source data is written by MCU. (PIO mode) 2'b11: Reserved mode.
4:0	DEC_TNUM	**indicates the correct capability in one block size.** 0: means the NFIECC is capable of correct 4 bits in one block size. 1: means the NFIECC is capable of correct 6 bits in one block size. 2: means the NFIECC is capable of correct 8 bits in one block size. 3: means the NFIECC is capable of correct 10 bits in one block size. 4: means the NFIECC is capable of correct 12 bits in one block size. 5: means the NFIECC is capable of correct 14 bits in one block size. 6: means the NFIECC is capable of correct 16 bits in one block size. 7: means the NFIECC is capable of correct 18 bits in one block size. 8: means the NFIECC is capable of correct 20 bits in one block size. 9: means the NFIECC is capable of correct 22 bits in one block size. 10: means the NFIECC is capable of correct 24 bits in one block size. 11: means the NFIECC is capable of correct 28 bits in one block size. 12: means the NFIECC is capable of correct 32 bits in one block size. 13: means the NFIECC is capable of correct 36 bits in one block size. 14: means the NFIECC is capable of correct 40 bits in one block size. 15: means the NFIECC is capable of correct 44 bits in one block size. 16: means the NFIECC is capable of correct 48 bits in one block size. 17: means the NFIECC is capable of correct 52 bits in one block size. 18: means the NFIECC is capable of correct 56 bits in one block size. 19: means the NFIECC is capable of correct 60 bits in one block size.

1100E108 NFIECC_DECDIADDR NFIECC Decoder DI Memory Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEC_DIADDR[29:14]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DEC_DIADDR[13:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Overview The register indicates the data start address of input data to the Decoder AHB mode.

Bit(s)	Name	Description
31:2	DEC_DIADDR	**indicates the memory address of input data to the Decoder block in AHB mode. (4-Byte align).**

1100E10C NFIECC_DECIDLE NFIECC Decoder Idle Status Register 0001

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DEC_IDLE
Type																RO
Reset																1

Overview This register indicates the Decoder Idle status.

Bit(s)	Name	Description
0	DEC_IDLE	**0: means the Decode block is under working.**

1: means the Decode block is in idle state and available for new coded block.

1100E110 NFIECC_DECFER NFIECC Decoder Found Error Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FER15	FER14	FER13	FER12	FER11	FER10	FER9	FER8	FER7	FER6	FER5	FER4	FER3	FER2	FER1	FER0
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview This register is for NFIECC Decoder status.

Bit(s)	Name	Description
15	FER15	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
14	FER14	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
13	FER13	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
12	FER12	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
11	FER11	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
10	FER10	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
9	FER9	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
8	FER8	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
7	FER7	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.

6	FER6	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
5	FER5	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
4	FER4	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
3	FER3	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
2	FER2	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
1	FER1	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.
0	FER0	**indicates the error found or not in the coded block. The FER numbered by NFI sector number in NFI mode, otherwise, in AHB mode, always use the FER0. The signal reset when DEC_EN is deasserted in both NFI and AHB mode.** 0: means there is no error detected in the coded block. 1: means there is(are) error(s) detected in the coded block.

1100E114 NFIECC_DECENUM0 NFIECC Decode Error Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ERRNUM3								ERRNUM2					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			ERRNUM1								ERRNUM0					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Overview The register indicates the error number of the coded block.

Bit(s)	Name	Description
29:24	ERRNUM3	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
21:16	ERRNUM2	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

13:8	ERRNUM1	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
5:0	ERRNUM0	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

1100E118 NFIECC_DECENUM1 NFIECC Decode Error Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ERRNUM7								ERRNUM6					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			ERRNUM5								ERRNUM4					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Overview The register indicates the error number of the coded block.

Bit(s)	Name	Description
29:24	ERRNUM7	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
21:16	ERRNUM6	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
13:8	ERRNUM5	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
5:0	ERRNUM4	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

1100E11C NFIECC_DECENUM2 NFIECC Decode Error Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ERRNUM11								ERRNUM10					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			ERRNUM9								ERRNUM8					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Overview The register indicates the error number of the coded block.

Bit(s)	Name	Description
29:24	ERRNUM11	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
21:16	ERRNUM10	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
13:8	ERRNUM9	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
5:0	ERRNUM8	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

1100E120 NFIECC_DECENUM3 NFIECC Decode Error Number Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			ERRNUM15								ERRNUM14					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			ERRNUM13								ERRNUM12					
Type			RO								RO					
Reset			0	0	0	0	0	0			0	0	0	0	0	0

Overview The register indicates the error number of the coded block.

Bit(s)	Name	Description
29:24	ERRNUM15	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
21:16	ERRNUM14	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
13:8	ERRNUM13	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.
5:0	ERRNUM12	**indicates the error numbers of coded block in one start signal. 6'h3f means the error is uncorrectable.** But ECC only can partially detect uncorrectable error. If the error number exceeds the error capability, ECC only can partially detect the situation.

1100E124 NFIECC_DECDONE NFIECC Decoder Error Status Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE	DONE

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Type	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview This register is for NFIECC Decoder done status.

Bit(s)	Name	Description
15	DONE15	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
14	DONE14	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
13	DONE13	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
12	DONE12	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
11	DONE11	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
10	DONE10	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
9	DONE9	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
8	DONE8	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
7	DONE7	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
6	DONE6	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
5	DONE5	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the

		definitions show in the Table 7.
4	DONE4	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
3	DONE3	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
2	DONE2	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
1	DONE1	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.
0	DONE0	**indicates the Decoding procedure is done.** 0: means the Decode block is under working. 1: means the Decode block is finished. For different DEC_CON and EMPTY_EN in NFIECC_DECCNFG register, decoder done has different meaning. Detail of the definitions show in the Table 7.

1100E128 NFIECC_DECEL0 NFIECC Decoder Error location0 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL1													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL0													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL1	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL0	**indicates the error location 1 of the decoding result.**

1100E12C NFIECC_DECEL1 NFIECC Decoder Error location1 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL3													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL2													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL3	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL2	**indicates the error location 1 of the decoding result.**

1100E130 NFIECC_DECEL2 NFIECC Decoder Error location2 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL5													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL4													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL5	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL4	**indicates the error location 1 of the decoding result.**

1100E134 NFIECC_DECEL3 NFIECC Decoder Error location3 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL7													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL6													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL7	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL6	**indicates the error location 1 of the decoding result.**

1100E138 NFIECC_DECEL4 NFIECC Decoder Error location4 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL9													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL8													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL9	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL8	**indicates the error location 1 of the decoding result.**

1100E13C NFIECC_DECEL5 NFIECC Decoder Error location5 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL11													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL10													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL11	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL10	**indicates the error location 1 of the decoding result.**

1100E140 NFIECC_DECEL6 NFIECC Decoder Error location6 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL13													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL12													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL13	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL12	**indicates the error location 1 of the decoding result.**

1100E144 NFIECC_DECEL7 NFIECC Decoder Error location7 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL15													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL14													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL15	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL14	**indicates the error location 1 of the decoding result.**

1100E148 NFIECC_DECEL8 NFIECC Decoder Error location8 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL17													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL16													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL17	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL16	**indicates the error location 1 of the decoding result.**

1100E14C NFIECC_DECEL9 NFIECC Decoder Error location9 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL19													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL18													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL19	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL18	**indicates the error location 1 of the decoding result.**

1100E150 NFIECC_DECEL10 NFIECC Decoder Error location10 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL21													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL20													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL21	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL20	**indicates the error location 1 of the decoding result.**

1100E154 NFIECC_DECEL11 NFIECC Decoder Error location11 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL23													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL22													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL23	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL22	**indicates the error location 1 of the decoding result.**

1100E158 NFIECC_DECEL12 NFIECC Decoder Error location12 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL25													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL24													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description

29:16	DEC_EL25	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL24	**indicates the error location 1 of the decoding result.**

1100E15C NFIECC_DECEL13 NFIECC Decoder Error location13 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL27													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL26													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL27	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL26	**indicates the error location 1 of the decoding result.**

1100E160 NFIECC_DECEL14 NFIECC Decoder Error location14 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL29													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL28													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL29	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL28	**indicates the error location 1 of the decoding result.**

1100E164 NFIECC_DECEL15 NFIECC Decoder Error location15 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL31													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL30													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL31	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL30	**indicates the error location 1 of the decoding result.**

1100E168 NFIECC_DECEL16 NFIECC Decoder Error location16 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL33													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL32													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL33	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL32	**indicates the error location 1 of the decoding result.**

1100E16C NFIECC_DECEL17 NFIECC Decoder Error location17 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL35													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL34													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL35	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL34	**indicates the error location 1 of the decoding result.**

1100E170 NFIECC_DECEL18 NFIECC Decoder Error location18 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL37													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name			DEC_EL36													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL37	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL36	**indicates the error location 1 of the decoding result.**

1100E174 NFIECC_DECEL19 NFIECC Decoder Error location19 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL39													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL38													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL39	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL38	**indicates the error location 1 of the decoding result.**

1100E178 NFIECC_DECEL20 NFIECC Decoder Error location20 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL41													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL40													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL41	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL40	**indicates the error location 1 of the decoding result.**

1100E17C NFIECC_DECEL21 NFIECC Decoder Error location21 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name			DEC_EL43													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL42													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL43	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL42	**indicates the error location 1 of the decoding result.**

1100E180 NFIECC_DECEL22 NFIECC Decoder Error location22 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL45													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL44													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL45	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL44	**indicates the error location 1 of the decoding result.**

1100E184 NFIECC_DECEL23 NFIECC Decoder Error location23 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL47													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL46													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL47	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL46	**indicates the error location 1 of the decoding result.**

1100E188 NFIECC_DECEL24 NFIECC Decoder Error location24 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL49													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL48													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL49	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL48	**indicates the error location 1 of the decoding result.**

1100E18C NFIECC_DECEL25 NFIECC Decoder Error location25 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL51													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL50													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL51	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL50	**indicates the error location 1 of the decoding result.**

1100E190 NFIECC_DECEL26 NFIECC Decoder Error location26 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL53													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL52													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL53	**indicates the error location 1 of the decoding result.**

13:0	DEC_EL52	**indicates the error location 1 of the decoding result.**

1100E194 NFIECC_DECEL27 NFIECC Decoder Error location27 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL55													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL54													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL55	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL54	**indicates the error location 1 of the decoding result.**

1100E198 NFIECC_DECEL28 NFIECC Decoder Error location28 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL57													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL56													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL57	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL56	**indicates the error location 1 of the decoding result.**

1100E19C NFIECC_DECEL29 NFIECC Decoder Error location29 Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name			DEC_EL59													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DEC_EL58													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overview The register indicates the error location of the decoding result

Bit(s)	Name	Description
29:16	DEC_EL59	**indicates the error location 1 of the decoding result.**
13:0	DEC_EL58	**indicates the error location 1 of the decoding result.**

1100E200 NFIECC_DECIRQEN NFIECC Decoder IRQ enable Register 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DEC_IRQEN
Type																RW
Reset																0

Overview This register is for software programmer to enable NFIECC IRQ signals (ignore in NFI mode)

Bit(s)	Name	Description
0	DEC_IRQEN	**Decoder IRQ mask: triggered when Decoder operation is completed.** 0: Disable 1: Enable

1100E204 NFIECC_DECIRQSTA NFIECC_DECIRQSTA 0000

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																DEC_IRQSTA
Type																RC
Reset																0

Overview This register is for software programmer tracking NFIECC IRQ status. (ignore in NFI mode)

Bit(s)	Name	Description
0	DEC_IRQSTA	**indicates Interrupt status for Decoder processing.** 0: No interrupt is generated. 1: An interrupt is pending and waiting for service. Active when Decoder processing is done.

1100E208 NFIECC_DECFSM NFIECC Decoder FSM 01010101

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		AUTOC_FSM											CHIEN_FSM			
Type		RO											RO			
Reset		0	0	0	0	0	0	1					0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					BMA_FSM								SYN_FSM			
Type					RO								RO			
Reset					0	0	0	1					0	0	0	1

Overview The register indicates the finite state machine status of decoder.

Bit(s)	Name	Description
30:24	AUTOC_FSM	**indicates the status of auto-correction stage.** 7'd1: IDLE 7'd2: COUNTE 7'd4: READ_PRE 7'd8: READ 7'd16: WRITE_PRE 7'd32: WRITE 7'd64: DONE
19:16	CHIEN_FSM	**indicates the status of Chien search stage.** 4'd1: IDLE 4'd2: BUSY 4'd4: DONE 4'd8: WAITC
11:8	BMA_FSM	**indicates the status of BMA stage.** 4'd1: IDLE 4'd2: EVENSYN 4'd4: BUSY 4'd8: DONE
3:0	SYN_FSM	**indicates the status of syndrome stage.** 4'd1: IDLE 4'd2: DATA 4'd4: DONE 4'd8: WAITBM

1100E20C NFIECC_BYPASS NFIECC BYPASS Async circuit 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BYPASS
Type																RW
Reset																0

Overview The register used to bypass the APB async design in NFIECC

Bit(s)	Name	Description
0	BYPASS	**The bit is used to bypass the APB async design in NFIECC. When set this bit to be1'b1, Ensure that NFIECC clock and APB clock are synchronous** 1'b0: means use async design in NFIECC 1'b1: means bypass the async design in NFIECC

15 Infrared Receiver

15.1 Introduction

This IR receiver can decode various IR transmission protocols. They could be divided into two groups. One is pulse-width coding such as NEC, REC-80 and SONY IR transmission protocol, the other is bi-phase coding, for example, Philips RC5, RC6.

15.2 Feature list

- Support max 56 bits length
- Support NEC, REC-80 and SONY IR transmission protocol
- Support Philips RC5, RC6 IR transmission protocol
- Support Philips RCMM IR transmission protocol
- HW key decoding
- HW pulse length decoding + SW key decoding
- HW key value detection for wake-up

15.3 Block Diagram

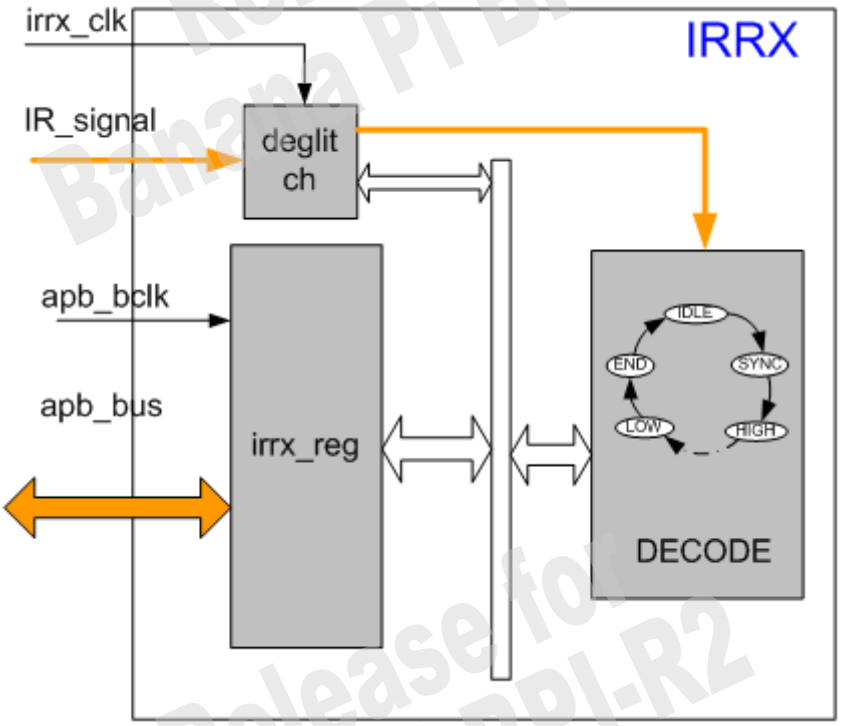


Figure 15-1: Block Diagram of IR Receiver

15.4 Register Definition

Module Base Address: (+10013000h)

ADDRESS	TITLE	DESCRIPTION
000 H	IRH	IR COUNT HIGH REGISTER
004 H	IRM	IR COUNT MIDIUM REGISTER
008 H	IRL	IR COUNT LOW REGISTER
00C H	IRCFGH	IR COUNT LOW REGISTER
010 H	IRCFGL	IR CONFIGURATION LOW
014 H	IRTHD	IR THRESHOLD REGISTER
018 H	IRRCM_THD	RCMM THRESHOLD REGISTER
01C H	IRRCM_THD_0	RCMM THRESHOLD REGISTER
020 H	IRCLR	IR CLEAR REGISTER
024 H	IREXP_EN	IR EXPECTATION REGISTER
028 H	EXP_BCNT	BITCNT EXPECTED VALUE REGISTER
02C H	ENEXP_IRM	IRM EXPECT VALUE BIT MASK REGISTER
030 H	ENEXP_IRL	IRM EXPECT VALUE BIT MASK REGISTER
034 H	EXP_IRL0	IRL EXPECT VALUE 0 REGISTER
038 H	EXP_IRL1	IRL EXPECT VALUE 1 REGISTER
03C H	EXP_IRL2	IRL EXPECT VALUE 2 REGISTER
040 H	EXP_IRL3	IRL EXPECT VALUE 3 REGISTER
044 H	EXP_IRL4	IRL EXPECT VALUE 4 REGISTER
048 H	EXP_IRL5	IRL EXPECT VALUE 5 REGISTER
04C H	EXP_IRL6	IRL EXPECT VALUE 6 REGISTER
050 H	EXP_IRL7	IRL EXPECT VALUE 7 REGISTER
054 H	EXP_IRL8	IRL EXPECT VALUE8 REGISTER
058 H	EXP_IRL9	IRL EXPECT VALUE9 REGISTER
05C H	EXP_IRM0	IRM EXPECT VALUE 0 REGISTER
060 H	EXP_IRM1	IRM EXPECT VALUE 1 REGISTER
064 H	EXP_IRM2	IRM EXPECT VALUE 2 REGISTER
068 H	EXP_IRM3	IRM EXPECT VALUE 3 REGISTER
06C H	EXP_IRM4	IRM EXPECT VALUE 4 REGISTER
070 H	EXP_IRM5	IRM EXPECT VALUE 5 REGISTER
074 H	EXP_IRM6	IRM EXPECT VALUE 6 REGISTER
078 H	EXP_IRM7	IRM EXPECT VALUE 7 REGISTER
07C H	EXP_IRM8	IRM EXPECT VALUE 8 REGISTER
080 H	EXP_IRM9	IRM EXPECT VALUE 9 REGISTER
084 H	PDWNCNT	POWER DOWN COUNTER REGISTER
088 H	CHK_DATA0	IR Pulse Width Length Value
08C H	CHK_DATA1	IR Pulse Width Length Value

090 H	CHK_DATA2	IR Pulse Width Length Value
094 H	CHK_DATA3	IR Pulse Width Length Value
098 H	CHK_DATA4	IR Pulse Width Length Value
09C H	CHK_DATA5	IR Pulse Width Length Value
0A0 H	CHK_DATA6	IR Pulse Width Length Value
0A4 H	CHK_DATA7	IR Pulse Width Length Value
0A8 H	CHK_DATA8	IR Pulse Width Length Value
0AC H	CHK_DATA9	IR Pulse Width Length Value
0B0 H	CHK_DATA10	IR Pulse Width Length Value
0B4 H	CHK_DATA11	IR Pulse Width Length Value
0B8 H	CHK_DATA12	IR Pulse Width Length Value
0BC H	CHK_DATA13	IR Pulse Width Length Value
0C0 H	CHK_DATA14	IR Pulse Width Length Value
0C4 H	CHK_DATA15	IR Pulse Width Length Value
0C8 H	CHK_DATA16	IR Pulse Width Length Value
0CC H	IRINT_EN	IR INTERRUPT ENABLE REGISTER
0D0 H	IR_INTCLR	PDWNC INTERRUPT CLEAR REGISTER
0D4 H	WDTSET	WDT MODE SET REGISTER
0D8 H	WDT	WDT COUNTER REGISTER
0DC H	WDTLMT	WDT LIMIT REGISTER
0E0 H	WDTLMT	WDT LIMIT REGISTER
0E4 H	POWKEY1	WDT POWER KEY REGISTER1
0E8 H	POWKEY2	WDT POWER KEY REGISTER2
0EC H	KEYMASK1	WDT MASK REGISTER1
0F0 H	KEYMASK1	WDT MASK REGISTER2

000 H IRH IR COUNT HIGH REGISTER 000000X0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RES_THREE								RES_TWO							
Type	R								R							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RES_ONE										BIT_CNT					
Type	R										R					
Reset	0	0	0	0	0	0	0	0			0	0	0	0	0	0

Bit Field	Name	Description
31:24	RES_THREE	Cycle Number of IR Clock of the 3rd BIT when RC5 decoding is disabled
23:16	RES_TWO	Cycle Number of IR Clock of the 2nd BIT when RC5 decoding is disabled
15:8	RES_ONE	Cycle Number of IR Clock of the 1st BIT when RC5 decoding is disabled
5:0	BIT_CNT	Present how many bit is decoded when RC5 decoding is disabled

004 H IRM IR COUNT MIDIUM REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BIT_REG3								BIT_REG2							
Type	R								R							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BIT_REG1								BIT_REG0							
Type	R								R							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	BIT_REG3	Decoded Bit Output Register 3
23:16	BIT_REG2	Decoded Bit Output Register 2
15:8	BIT_REG1	Decoded Bit Output Register 1
7:0	BIT_REG0	Decoded Bit Output Register 0

008 H IRL IR COUNT LOW REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									BIT_REG6							
Type									R							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BIT_REG5								BIT_REG4							
Type	R								R							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	BIT_REG6	Decoded Bit Output Register 6
15:8	BIT_REG5	Decoded Bit Output Register 5
7:0	BIT_REG4	Decoded Bit Output Register 4

00C H IRCFGH IR COUNT LOW REGISTER XXXFXX00

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										OK_PERIOD						
Type										R/W						
Reset										0	0	0	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DISPD	IGB0	CHK_EN						DISH	DISL	IGN_1ST	ORDINV	RC5_1ST	RC5	IRINV	IREN
Type	R/W	R/W	R/W						R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0						0	0	0	0	0	0	0	0

Bit Field	Name	Description
22:16	OK_PERIOD	IR end patterns length When the value of the sampling counter is IREND+1, the IR end signal is active. Then, IR decoder completes this IR command.
15	DISPD	IR clear disable at power down mode 0: IR state machine will be auto-cleared when IC is at power saving mode. 1: IR state machine will be not auto-cleared when IC is at power saving mode.
14	IGB0	Ignore bit 0 Ignore 0 bit count IR pulse in order to reduce noise disturbance. 0: Ignore 0 bit count disable 1: Ignore 0 bit count enable
13	CHK_EN	Enable IR pulse width detection
7	DISH	Disable high Disable sampling counter when IR high.
6	DISL	Disable low Disable sampling counter when IR low.
5	IGN_1ST	Ignore synchronization pulse The first transition is viewed as synchronization pulse, and isn't recorded.
4	ORDINV	Order inverse. The decoded IR pulse is bit-reversed. 0: Keep the order of decoded bit 1: Inverse the order of decoded bit
3	RC5_1ST	RC5 first RC5 first bit will also be decoded, but the decoded value will be bit-reversed. If this bit is disabled, the first bit of RC5 will be treated as sync pulse and will not be decoded.
2	RC5	RC5 Use RC5 format.
1	IRINV	IR inverse The IR pulse is inversed before decoded.
0	IREN	IR Enable Enable IR hardware receiver function.

010 H IRCFGL IR CONFIGURATION LOW XXXXXXFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name												CHK_PERIOD[12:8]				
Type												R/W				
Reset												0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_PERIOD[7:0]								SAPERIOD							
Type	R/W								R/W							
Reset									1	1	1	1	1	1	1	1

Bit Field	Name	Description

20:8	CHK_PERIOD	Check IR pulse width sampling period This field decides the working frequency of the sampling counter for IR pulse width detection. Period is CHK_PERIOD[10:0] * PERIOD of Bus Clock.
7:0	SAPERIOD	Sampling period This field decides the working frequency of the sampling counter. Period is SA_PERIOD[7:0] * PERIOD of IR Clock.

014 H IRTHD IR THRESHOLD REGISTER XXXXX600

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				**DG_CNT**					**ICLR**	**IRTHD**						
Type				R/W					R/W	R/W						
Reset				0	0	1	1	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
12:8	DG_CNT	De-glitch counter. When IR signal stays at "High" or "Low" shorter than time defined by DE_CNT, it will be viewed as "glitch" and be ignored.
7	ICLR	Interrupt clear reset IR Activate this bit to clear IR state machine and IRH/IRM/IRL when users clear IR interrupt by IR_INTCLR.
6:0	IRTHD	IR threshold When RC5 decoding is disabled and IR signal stays "High" longer than IRTHD[6:0] "IR Clock Period, the decoder will output bit "One".

018 H IRRCM_THD RCMM THRESHOLD REGISTER X0000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**RCMM**				**IRRCM_THD_11**							**IRRCM_THD_10[6:2]**				
Type	R/W				R/W							R/W				
Reset	0				0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**IRRCM_THD_10[1:0]**		**IRRCM_THD_01**							**IRRCM_THD_00**						
Type	R/W		R/W							R/W						
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31	RCMM	RCMM enable 0: disable rcmm 1: enable rcmm
27:21	IRRCM_THD_11	IR rcmm threshold

		When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_10[6:0] and shorter than IRRCM_THD11, the decoder will output bit "01".
20:14	IRRCM_THD_10	IR rcmm threshold When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_10[6:0] and shorter than IRRCM_THD11, the decoder will output bit "01".
13:7	IRRCM_THD_01	IR rcmm threshold When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_00[6:0] and short than IRRCM_THD01, the decoder will output bit "00".
6:0	IRRCM_THD_00	IR rcmm threshold When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_00[6:0] and short than IRRCM_THD01, the decoder will output bit "00".

01C H IRRCM_THD_0 RCMM THRESHOLD REGISTER XXXXX000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			IRRCM_THD_21							IRRCM_THD_20						
Type			R/W							R/W						
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
13:7	IRRCM_THD_21	IR rcmm threshold When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_20[6:0] and short than IRRCM_THD21, the decoder will output bit "10".
6:0	IRRCM_THD_20	IR rcmm threshold When RCMM decoding is ENABLE and IR signal stays "High" longer than IRTHD_20[6:0] and short than IRRCM_THD21, the decoder will output bit "10".

020 H IRCLR IR CLEAR REGISTER XXXXXXXX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																IRCLR
Type																W
Reset																0

Bit Field	Name	Description
0	IRCLR	IR clear Clear IR state machine and IRH/IRM/IRL. Before IRRX decode input IR signal, CPU has to write this register.

024 H IREXP_EN IR EXPECTATION REGISTER XXXXX000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					PD_IREXPEN		IRPDWN_EN	BCEPEN	IREXPEN							
Type					R/W		R/W	R/W	R/W							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
11:10	PD_IREXPEN	Power down expectation enable We have two expected IR code to enter power saving mode. If enabled, system will enter power saving mode when received IR matched expectation value stored in the register.
9	IRPDWN_EN	IR power down function enable
8	BCEPEN	Bit count expectation enable If enabled, PDWNC module will wake up whole system when the number of bit of received IR is equal to EXP_BCNT register.
7:0	IREXPEN	Expectation enable If enabled, PDWNC module will wake up whole system when received IR is equal to the expectation value stored in the corresponding register.

028 H EXP_BCNT BITCNT EXPECTED VALUE REGISTER XXXXX000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DAT_CLR_STS															
Type	R															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				chk_cnt							EXP_BITCNT					
Type				R							R/W					
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description

31	DAT_CLR_STS	ir data clear status, when write clear ir data, can polling this bit status, when is 1, indicate data not clear, when this bit status is 0, indicate ir data clear done
12:6	chk_cnt	read only .for pulse counter
5:0	EXP_BITCNT	Expect bit count Expected IR power down wake up bit count value

02C H ENEXP_IRM IRM EXPECT VALUE BIT MASK REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ENEXP_BIT_REG3								ENEXP_BIT_REG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENEXP_BIT_REG1								ENEXP_BIT_REG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	ENEXP_BIT_REG3	Enable expected value byte 3 Expected IR power down wake up code BITREG3 enable bits
23:16	ENEXP_BIT_REG2	Enable expected value byte 2 Expected IR power down wake up code BITREG2 enable bits
15:8	ENEXP_BIT_REG1	Enable expected value byte 1 Expected IR power down wake up code BITREG1 enable bits
7:0	ENEXP_BIT_REG0	Enable expected value byte 0 Expected IR power down wake up code BITREG0 enable bits

030 H ENEXP_IRL IRM EXPECT VALUE BIT MASK REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									ENEXP_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ENEXP_BITREG5								ENEXP_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	ENEXP_BITREG6	Enable expected value byte 6 Expected IR power down wake up code BITREG6 enable bits
15:8	ENEXP_BITREG5	Enable expected value byte 5 Expected IR power down wake up code BITREG5 enable bits
7:0	ENEXP_BITREG4	Enable expected value byte 4 Expected IR power down wake up code BITREG4 enable bits

034 H EXP_IRL0 IRL EXPECT VALUE 0 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP0_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP0_BITREG5								EXP0_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP0_BITREG6	The expected value 0 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP0_BITREG5	The expected value 0 byte 5 Expected IR power down wake up code BITREG6
7:0	EXP0_BITREG4	The expected value 0 byte 4 Expected IR power down wake up code BITREG4

038 H EXP_IRL1 IRL EXPECT VALUE 1 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP1_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP1_BITREG5								EXP1_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP1_BITREG6	The expected value 1 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP1_BITREG5	The expected value 1 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP1_BITREG4	The expected value 1 byte 4 Expected IR power down wake up code BITREG4

03C H EXP_IRL2 IRL EXPECT VALUE 2 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP2_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	EXP2_BITREG5								EXP2_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP2_BITREG6	The expected value 2 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP2_BITREG5	The expected value 2 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP2_BITREG4	The expected value 2 byte 4 Expected IR power down wake up code BITREG4

040 H EXP_IRL3 IRL EXPECT VALUE 3 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP3_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP3_BITREG5								EXP3_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP3_BITREG6	The expected value 3 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP3_BITREG5	The expected value 3 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP3_BITREG4	The expected value 3 byte 4 Expected IR power down wake up code BITREG4

044 H EXP_IRL4 IRL EXPECT VALUE 4 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP4_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP4_BITREG5								EXP4_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP4_BITREG6	The expected value 4 byte 6

		Expected IR power down wake up code BITREG6
15:8	EXP4_BITREG5	The expected value 4 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP4_BITREG4	The expected value 4 byte 4 Expected IR power down wake up code BITREG4

048 H EXP_IRL5 IRL EXPECT VALUE 5 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP5_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP5_BITREG5								EXP5_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP5_BITREG6	The expected value 0 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP5_BITREG5	The expected value 0 byte 5 Expected IR power down wake up code BITREG6
7:0	EXP5_BITREG4	The expected value 0 byte 4 Expected IR power down wake up code BITREG4

04C H EXP_IRL6 IRL EXPECT VALUE 6 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP6_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP6_BITREG5								EXP6_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP6_BITREG6	The expected value 1 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP6_BITREG5	The expected value 1 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP6_BITREG4	The expected value 1 byte 4 Expected IR power down wake up code BITREG4

050 H EXP_IRL7 IRL EXPECT VALUE 7 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP7_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP7_BITREG5								EXP7_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP7_BITREG6	The expected value 2 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP7_BITREG5	The expected value 2 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP7_BITREG4	The expected value 2 byte 4 Expected IR power down wake up code BITREG4

054 H EXP_IRL8 IRL EXPECT VALUE8 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP8_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP8_BITREG5								EXP8_BITREG4							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP8_BITREG6	The expected value 3 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP8_BITREG5	The expected value 3 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP8_BITREG4	The expected value 3 byte 4 Expected IR power down wake up code BITREG4

058 H EXP_IRL9 IRL EXPECT VALUE9 REGISTER XX000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									EXP9_BITREG6							
Type									R/W							
Reset									0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP9_BITREG5								EXP4_BITREG4							
Type	R/W								R/W							

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
23:16	EXP9_BITREG6	The expected value 4 byte 6 Expected IR power down wake up code BITREG6
15:8	EXP9_BITREG5	The expected value 4 byte 5 Expected IR power down wake up code BITREG5
7:0	EXP4_BITREG4	The expected value 4 byte 4 Expected IR power down wake up code BITREG4

05C H EXP_IRM0 IRM EXPECT VALUE 0 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP0_BITREG3								EXP0_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP0_BITREG1								EXP0_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP0_BITREG3	The expected value 0 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP0_BITREG2	The expected value 0 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP0_BITREG1	The expected value 0 byte 1 Expected IR power down wake up code BITREG2
7:0	EXP0_BITREG0	The expected value 0 byte 0 Expected IR power down wake up code BITREG0

060 H EXP_IRM1 IRM EXPECT VALUE 1 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP1_BITREG3								EXP1_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP1_BITREG1								EXP1_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP1_BITREG3	The expected value 1 byte 3

		Expected IR power down wake up code BITREG3
23:16	EXP1_BITREG2	The expected value 1 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP1_BITREG1	The expected value 1 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP1_BITREG0	The expected value 1 byte 0 Expected IR power down wake up code BITREG0

064 H EXP_IRM2 IRM EXPECT VALUE 2 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP2_BITREG3								EXP2_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP2_BITREG1								EXP2_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP2_BITREG3	The expected value 2 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP2_BITREG2	The expected value 2 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP2_BITREG1	The expected value 2 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP2_BITREG0	The expected value 2 byte 0 Expected IR power down wake up code BITREG0

068 H EXP_IRM3 IRM EXPECT VALUE 3 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP3_BITREG3								EXP3_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP3_BITREG1								EXP3_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP3_BITREG3	The expected value 3 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP3_BITREG2	The expected value 3 byte 2 Expected IR power down wake up code BITREG2

15:8	EXP3_BITREG1	The expected value 3 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP3_BITREG0	The expected value 3 byte 0 Expected IR power down wake up code BITREG0

06C H EXP_IRM4 IRM EXPECT VALUE 4 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP4_BITREG3								EXP4_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP4_BITREG1								EXP4_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP4_BITREG3	The expected value 4 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP4_BITREG2	The expected value 4 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP4_BITREG1	The expected value 4 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP4_BITREG0	The expected value 4 byte 0 Expected IR power down wake up code BITREG0

070 H EXP_IRM5 IRM EXPECT VALUE 5 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP5_BITREG3								EXP5_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP5_BITREG1								EXP5_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP5_BITREG3	The expected value 5 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP5_BITREG2	The expected value 0 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP5_BITREG1	The expected value 0 byte 1 Expected IR power down wake up code BITREG2
7:0	EXP5_BITREG0	The expected value 0 byte 0 Expected IR power down wake up code BITREG0

074 H EXP_IRM6 IRM EXPECT VALUE 6 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP6_BITREG3								EXP6_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP6_BITREG1								EXP6_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP6_BITREG3	The expected value 6 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP6_BITREG2	The expected value 1 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP6_BITREG1	The expected value 1 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP6_BITREG0	The expected value 1 byte 0 Expected IR power down wake up code BITREG0

078 H EXP_IRM7 IRM EXPECT VALUE 7 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP7_BITREG3								EXP7_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP7_BITREG1								EXP7_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP7_BITREG3	The expected value 7 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP7_BITREG2	The expected value 2 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP7_BITREG1	The expected value 2 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP7_BITREG0	The expected value 2 byte 0 Expected IR power down wake up code BITREG0

07C H EXP_IRM8 IRM EXPECT VALUE 8 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	EXP8_BITREG3								EXP8_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP8_BITREG1								EXP8_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP8_BITREG3	The expected value 8 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP8_BITREG2	The expected value 3 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP8_BITREG1	The expected value 3 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP8_BITREG0	The expected value 3 byte 0 Expected IR power down wake up code BITREG0

080 H EXP_IRM9 IRM EXPECT VALUE 9 REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXP9_BITREG3								EXP9_BITREG2							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	EXP9_BITREG1								EXP4_BITREG0							
Type	R/W								R/W							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:24	EXP9_BITREG3	The expected value 9 byte 3 Expected IR power down wake up code BITREG3
23:16	EXP9_BITREG2	The expected value 4 byte 2 Expected IR power down wake up code BITREG2
15:8	EXP9_BITREG1	The expected value 4 byte 1 Expected IR power down wake up code BITREG1
7:0	EXP4_BITREG0	The expected value 4 byte 0 Expected IR power down wake up code BITREG0

084 H PDWNCNT POWER DOWN COUNTER REGISTER 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDWNCNT[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDWNCNT[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	PDWNCNT	Power down counter This register decides how long the system enters power saving mode.

088 H CHK_DATA0 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA0[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA0[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA0	IR 1-4 pulse width length value, each pulse length is 1byte

08C H CHK_DATA1 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA1[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA1[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA1	IR 5-8 pulse width length value, each pulse length is 1byte

090 H CHK_DATA2 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA2[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA2[15:0]															

Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA2	IR 9-12 pulse width length value, each pulse length is 1byte

094 H CHK_DATA3 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA3[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA3[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA3	IR 13-16 pulse width length value, each pulse length is 1byte

098 H CHK_DATA4 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA4[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA4[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA4	IR 17-20 pulse width length value, each pulse length is 1byte

09C H CHK_DATA5 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA5[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA5[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA5	IR 21-24 pulse width length value, each pulse length is 1byte

0A0 H CHK_DATA6 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA6[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA6[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA6	IR 25-28 pulse width length value, each pulse length is 1byte

0A4 H CHK_DATA7 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA7[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA7[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA7	IR 29-32 pulse width length value, each pulse length is 1byte

0A8 H CHK_DATA8 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA8[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA8[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Name	Description

Field		
31:0	CHK_DATA8	IR 33-36 pulse width length value, each pulse length is 1byte

0AC H CHK_DATA9 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA9[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA9[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA9	IR 37-40 pulse width length value, each pulse length is 1byte

0B0 H CHK_DATA10 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA10[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA10[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA10	IR 41-44 pulse width length value, each pulse length is 1byte

0B4 H CHK_DATA11 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA11[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA11[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA11	IR 45-48 pulse width length value, each pulse length is 1byte

0B8 H CHK_DATA12 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA12[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA12[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA12	IR 49-52 pulse width length value, each pulse length is 1byte

0BC H CHK_DATA13 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA13[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA13[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA13	IR 53-56 pulse width length value, each pulse length is 1byte

0C0 H CHK_DATA14 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA14[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA14[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA14	IR 57-60 pulse width length value, each pulse length is 1byte

0C4 H CHK_DATA15 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA15[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA15[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA15	IR 61-64 pulse width length value, each pulse length is 1byte

0C8 H CHK_DATA16 IR Pulse Width Length Value 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CHK_DATA16[31:16]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CHK_DATA16[15:0]															
Type	R/W															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
31:0	CHK_DATA16	IR 65-68 pulse width length value, each pulse length is 1byte

0CC H IRINT_EN IR INTERRUPT ENABLE REGISTER XXXXXXXX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																IRINT_EN
Type																R/W
Reset																0

Bit Field	Name	Description
0	IRINT_EN	IR receiver interrupt enable bit

0D0 H IR_INTCLR PDWNC INTERRUPT CLEAR REGISTER XXXXXXXX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																

Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																**IR_INT CLR**
Type																R/W
Reset																0

Bit Field	Name	Description
0	IR_INTCLR	IR receiver interrupt clear bit

0D4 H WDTSET WDT MODE SET REGISTER XXXXXXXX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										**WDTMODE**					**DBG_ STOP**	**WDT_ EN**
Type										R/W					R/W	R/W
Reset										0					0	0

Bit Field	Name	Description
6:4	WDTMODE	
1	DBG_STOP	Stop Watch Dog Timer while RISC in ICE debug mode This Register can only be accessed by ARM
0	WDT_EN	Watch Dog Timer Enable

0D8 H WDT WDT COUNTER REGISTER XXXXXXXX

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**WDT[31:16]**															
Type	R/W															
Reset	0															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**WDT[15:0]**															
Type	R/W															
Reset																

Bit Field	Name	Description
31:0	WDT	Watch Dog Timer value, while enabled, the timer increase by 1 by 3M clock This Register can only be accessed by ARM

0DC H WDTLMT WDT LIMIT REGISTER FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WDTLMT[31:16]															
Type	R/W															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	WDTLMT[15:0]															
Type	R/W															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit Field	Name	Description
31:0	WDTLMT	Watch Dog Timer Limit Register This Register can only be accessed by ARM

0E0 H WDTLMT WDT LIMIT REGISTER XXXXX000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					WDT_TRIG	WDT_SMP	WDTRST_MODE		WDTRST_KEY							
Type					R/W	R/W	R/W		R/W							
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit Field	Name	Description
11	WDT_TRIG	Set 1 if watch dog timer ever reset the system This Register can only be accessed by ARM
10	WDT_SMP	If watch dog timer ever reset the system, sample the signal wdt_assert, usually 1 This Register can only be accessed by ARM
9:8	WDTRST_MODE	Last system reset by which watch do timer mode This Register can only be accessed by ARM
7:0	WDTRST_KEY	Byte exclusive of received IR key [55:0] when WDT trigger This Register can only be accessed by ARM

0E4 H POWKEY1 WDT POWER KEY REGISTER1 5A5A5A5A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	POWKEY1[31:16]															
Type	R/W															
Reset	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	POWKEY1[15:0]															

Type	R/W															
Reset	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

Bit Field	Name	Description
31:0	POWKEY1	Watch Dog Timer Limit Register This Register can only be accessed by ARM

0E8 H POWKEY2 WDT POWER KEY REGISTER2 XX5A5A5A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									**POWKEY2[23:16]**							
Type									R/W							
Reset									0	1	0	1	1	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**POWKEY2[15:0]**															
Type	R/W															
Reset	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

Bit Field	Name	Description
23:0	POWKEY2	Watch Dog Timer Power Key register[55:32] This Register can only be accessed by ARM

0EC H KEYMASK1 WDT MASK REGISTER1 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	**KEYMASK1[31:16]**															
Type	R/W															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**KEYMASK1[15:0]**															
Type	R/W															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit Field	Name	Description
31:0	KEYMASK1	Watch Dog Timer Power Key Mask register[31:0], set to 1 indicated this bit is to be matched This Register can only be accessed by ARM

0F0 H KEYMASK1 WDT MASK REGISTER2 XXFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									**KEYMASK2[23:16]**							
Type									R/W							
Reset									1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	KEYMASK2[15:0]															
Type	R/W															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit Field	Name	Description
23:0	KEYMASK2	Watch Dog Timer Power Key Mask register[55:32], set to 1 indicated this bit is to be matched This Register can only be accessed by ARM

16 HDMI Transmitter

16.1 Introduction

The HDMI Transmitter is used to generate HDMI format data with HDMI Specification 1.4a.

16.2 Feature list

- HDMI deepcolor support 16bit
- Audio support multi-channel input and spdif input
- HDCP encrypt video data and audio data
- Info-frame packets can programmed by soft-ware

16.3 Block Diagram

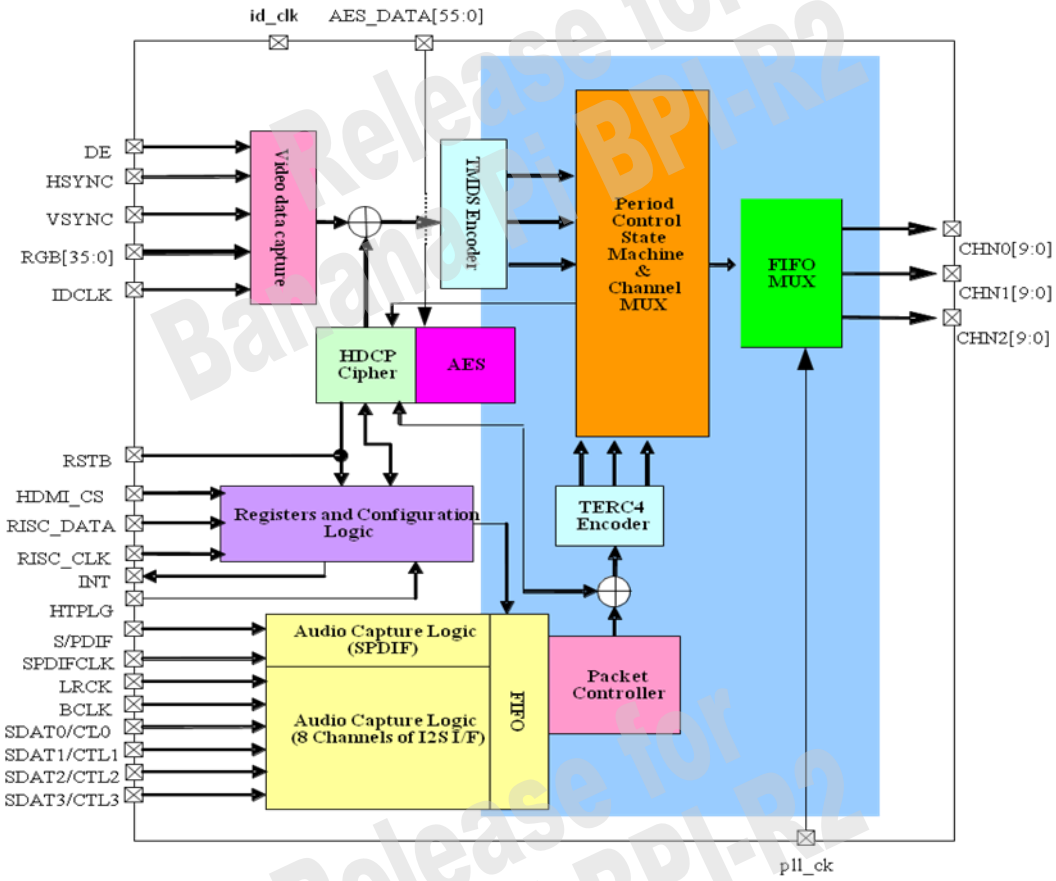


Figure 16-1: Block Diagram of HDMI Transmitter

16.4 Register Definition

Module name: HDMI_SHELL Base address: (+14015000h)

Address	Name	Width	Register Function
14015000	VEN_IDL	8	**Vendor ID** This register contains vendor ID low 8 bit.
14015004	VEN_IDH	8	**Vendor ID** This register contains vendor ID high 8 bit.
14015008	DEV_IDL	8	**Device ID** This register contains device ID low 8 bit.
1401500C	DEV_IDH	8	**Device ID** This register contains device ID high 8 bit.
14015010	DEV_REV	8	**Revision ID** This register contains the revision ID.
14015014	INT	8	**Interrupt** This register contains the interrupt of hdmi_shell
14015018	INT_MASK	8	**Int mask** This register contains the interrupt mask of hdmi_shell
1401501C	CTRL	8	**CTRL**
14015020	STATUS	8	**Status**
14015024	CFG0	8	**Config register 0**
14015028	CFG1	8	**Config register 1**
1401502C	CFG2	8	**Config register 2**
14015030	CFG3	8	**Config register 3**
14015034	CFG4	8	**Config register 4**
14015038	CFG5	8	**Config register 5**
14015040	WR_BKSV0	8	**Write for BKSV[7:0]** These 5 bytes of register set allows firmware write 5 bytes of BKSV read from receiver.
14015044	WR_BKSV1	8	**Write for BKSV[15:8]** These 5 bytes of register set allows firmware write 5 bytes of BKSV read from receiver.
14015048	WR_BKSV2	8	**Write for BKSV[23:16]** These 5 bytes of register set allows firmware write 5 bytes of BKSV read from receiver.
1401504C	WR_BKSV3	8	**Write for BKSV[31:24]** These 5 bytes of register set allows firmware write 5 bytes of BKSV read from receiver.
14015050	WR_BKSV4	8	**Write for BKSV[39:32]** These 5 bytes of register set allows firmware write 5 bytes of BKSV read from receiver.
14015054	WR_AN0	8	**Write for AN[7:0]** This 8-byte register set contains 64 bits pseudo-random value.
14015058	WR_AN1	8	**Write for AN[15:8]** This 8-byte register set contains 64 bits pseudo-random value.
1401505C	WR_AN2	8	**Write for AN[23:16]** This 8-byte register set contains 64 bits pseudo-random value.
14015060	WR_AN3	8	**Write for AN[31:24]** This 8-byte register set contains 64 bits pseudo-random value.

14015064	**WR_AN4**	8	**Write for AN[39:32]** This 8-byte register set contains 64 bits pseudo-random value.
14015068	**WR_AN5**	8	**Write for AN[47:40]** This 8-byte register set contains 64 bits pseudo-random value.
1401506C	**WR_AN6**	8	**Write for AN[55:48]** This 8-byte register set contains 64 bits pseudo-random value.
14015070	**WR_AN7**	8	**Write for AN[63:56]** This 8-byte register set contains 64 bits pseudo-random value.
14015074	**RD_AKSV0**	8	**Read for AKSV[7:0]** These 5 bytes of register set allows firmware Read 5 bytes of AKSV from receiver.
14015078	**RD_AKSV1**	8	**Read for AKSV[15:8]** These 5 bytes of register set allows firmware Read 5 bytes of AKSV from receiver.
1401507C	**RD_AKSV2**	8	**Read for AKSV[23:16]** These 5 bytes of register set allows firmware Read 5 bytes of AKSV from receiver.
14015080	**RD_AKSV3**	8	**Read for AKSV[31:24]** These 5 bytes of register set allows firmware Read 5 bytes of AKSV from receiver.
14015084	**RD_AKSV4**	8	**Read for AKSV[39:32]** These 5 bytes of register set allows firmware Read 5 bytes of AKSV from receiver.
14015088	**RI_0**	8	**RI[7:0]** This 2-byte read only register contains the synchronization verification value Ri.
1401508C	**RI_1**	8	**RI[15:8]** This 2-byte read only register contains the synchronization verification value Ri.
14015090	**KEY_PORT**	8	**Key port** This port is used by host to supply key to transmitter for HDCP authentication.
14015094	**KSVKIST**	8	**KSV lisy** This port is used by host to supply KSV list to transmitter to compute the KSV list integrity verification value for HDCP repeater authentication.
14015098	**I2CM_ADDR**	8	**I2C Addr** This register allows host to program I2C device ADDR8 of the EEPROM that contains the KSV and key set for HDCP authentication process. The default value after Power_On reset is defined from the PAD_GPO_3 and PAD_GPO_4 as following:
1401509C	**BIST0**	8	**BIST[7:0]** This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.
140150A0	**BIST1**	8	**BIST[15:8]** This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.
140150A4	**BIST2**	8	**BIST[23:16]** This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.
140150A8	**BIST3**	8	**BIST[29:24]** This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.
140150B8	**HDCP_STA**	8	**HDCP status** This register contains the status of hdcp
140150BC	**HDCP_CTRL**	8	**HDCP control**

			This register control the process of the hdcp
140150C0	**V0**	8	**Repeater Hash V[7:0]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150C4	**V1**	8	**Repeater Hash V[15:8]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150C8	**V2**	8	**Repeater Hash V[23:16]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150CC	**V3**	8	**Repeater Hash V[31:24]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150D0	**V4**	8	**Repeater Hash V[39:32]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150D4	**V5**	8	**Repeater Hash V[47:40]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150D8	**V6**	8	**Repeater Hash V[55:48]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150DC	**V7**	8	**Repeater Hash V[63:56]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150E0	**V8**	8	**Repeater Hash V[71:64]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150E4	**V9**	8	**Repeater Hash V[79:72]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150E8	**V10**	8	**Repeater Hash V[87:80]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150EC	**V11**	8	**Repeater Hash V[95:88]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150F0	**V12**	8	**Repeater Hash V[103:96]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150F4	**V13**	8	**Repeater Hash V[111:104]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150F8	**V14**	8	**Repeater Hash V[119:112]** This port is used by host to write KSV list integrity verification value computed by repeater.
140150FC	**V15**	8	**Repeater Hash V[127:120]** This port is used by host to write KSV list integrity verification value computed by repeater.
14015100	**V16**	8	**Repeater Hash V[135:128]** This port is used by host to write KSV list integrity verification value computed by repeater.
14015104	**V17**	8	**Repeater Hash V[143:136]** This port is used by host to write KSV list integrity verification value computed by repeater.
14015108	**V18**	8	**Repeater Hash V[151:144]**

			This port is used by host to write KSV list integrity verification value computed by repeater.
1401510C	V19	8	**Repeater Hash V[159:152]** This port is used by host to write KSV list integrity verification value computed by repeater.
14015154	I2S_UV	8	**I2S_UV**
14015158	CTRL_1	8	**Info autosend ctrl** This register can control some info autosend on or off
1401515C	STATUS_1	8	**Info status** This register contains some info status
14015160	CTS_CTRL	8	**Soft control NCTS** This register is used by host microcontroller to select the source for CTS value.
14015164	CTS0	8	**CTS[7:0]** These three read only registers contain the CTS value generated by hardware.
14015168	CTS1	8	**CTS[15:8]** These three read only registers contain the CTS value generated by hardware.
1401516C	CTS2	8	**CTS[19:16]** These three read only registers contain the CTS value generated by hardware.
14015178	DIV_RESET	8	
1401517C	AUDIO_CFG	8	
14015180	SRAM_DELSEL	8	**SRAM delay sel** This register contains the delay cycle for Sram
14015184	NCTS	8	**Write for N,CTS** These registers are implemented in a register file where an I2C ADDR8 is assigned to its data port.
14015188	IFM_PORT	8	**Information Port** This data port is used by host to supply InfoFrame.
1401518C	CH_SWITCH0	8	**CH_SWITCH[7:0]**
14015190	CH_SWITCH1	8	**CH_SWITCH[15:8]**
14015194	CH_SWITCH2	8	**CH_SWITCH[23:16]**
14015198	LR_SWITCH	8	**LR_SWITCH**
1401519C	InfoFrame_Header_1	8	**InfoFrame version**
140151A0	InfoFrame_Header_2	8	**InfoFrame type**
140151A4	InfoFrame_Header_3	8	**InfoFrame length**
140151B0	Shift_length_R2	8	**Shift_length_R2**
140151B4	Mix_Ctrl	8	**Mix_Ctrl**
140151B8	IIR_Filter	8	**IIR_Filter** This register is the port of IIR filter coef
140151BC	Shift_length_L0	8	**Shift_length_L0**
140151C0	Shift_length_L1	8	**Shift_length_L1**
140151C4	AOUT_CFG	8	This register contains the audio out configration
14015200	L_STATUS_0	8	**L_STATUS[7:0]**

14015204	L_STATUS_1	8	L_STATUS[15:8]
14015208	L_STATUS_2	8	L_STATUS[23:16]
1401520C	L_STATUS_3	8	L_STATUS[31:23]
14015210	L_STATUS_4	8	L_STATUS[39:32]
14015214	L_STATUS_5	8	L_STATUS[47:40]
14015218	L_STATUS_6	8	L_STATUS6[55:48]
1401521C	L_STATUS_7	8	L_STATUS[63:56]
14015220	L_STATUS_8	8	L_STATUS[71:64]
14015224	L_STATUS_9	8	L_STATUS[79:72]
14015228	L_STATUS_10	8	L_STATUS[87:80]
1401522C	L_STATUS_11	8	L_STATUS1[95:88]
14015230	L_STATUS_12	8	L_STATUS[103:96]
14015234	L_STATUS_13	8	L_STATUS[111:104]
14015238	L_STATUS_14	8	L_STATUS[119:112]
1401523C	L_STATUS_15	8	L_STATUS[127:120]
14015240	L_STATUS_16	8	L_STATUS[135:128]
14015244	L_STATUS_17	8	L_STATUS[143:136]
14015248	L_STATUS_18	8	L_STATUS[151:144]
1401524C	L_STATUS_19	8	L_STATUS[159:152]
14015250	L_STATUS_20	8	L_STATUS[167:160]
14015254	L_STATUS_21	8	L_STATUS[175:168]
14015258	L_STATUS_22	8	L_STATUS[183:176]
1401525C	L_STATUS_23	8	L_STATUS[191:184]
14015260	R_STATUS_0	8	R_STATUS[7:0]
14015264	R_STATUS_1	8	R_STATUS[15:8]
14015268	R_STATUS_2	8	R_STATUS[23:16]
1401526C	R_STATUS_3	8	R_STATUS[31:24]
14015270	R_STATUS_4	8	R_STATUS[39:32]
14015274	R_STATUS_5	8	R_STATUS[47:40]
14015278	R_STATUS_6	8	R_STATUS[55:48]
1401527C	R_STATUS_7	8	R_STATUS[63:56]
14015280	R_STATUS_8	8	R_STATUS[71:64]
14015284	R_STATUS_9	8	R_STATUS[79:72]
14015288	R_STATUS_10	8	R_STATUS[87:80]
1401528C	R_STATUS_11	8	R_STATUS[95:88]
14015290	R_STATUS_12	8	R_STATUS[103:96]
14015294	R_STATUS_13	8	R_STATUS[111:104]

14015298	R_STATUS_14	8	R_STATUS[119:112]
1401529C	R_STATUS_15	8	R_STATUS[127:120]
140152A0	R_STATUS_16	8	R_STATUS[135:128]
140152A4	R_STATUS_17	8	R_STATUS[143:126]
140152A8	R_STATUS_18	8	R_STATUS[151:144]
140152AC	R_STATUS_19	8	R_STATUS[159:152]
140152B0	R_STATUS_20	8	R_STATUS[167:160]
140152B4	R_STATUS_21	8	R_STATUS[175:168]
140152B8	R_STATUS_22	8	R_STATUS[183:176]
140152BC	R_STATUS_23	8	R_STATUS[191:184]
140152C0	VOLUME0	8	VOLUME_TARGET[7:0]
140152C4	VOLUME1	8	VOLUME_TARGET[15:8]
140152C8	VOLUME2	8	VOLUME_TARGET[23:16]
140152CC	VOLUME3	8	VOLUME_STEP[7:0]
140152D0	VOLUME4	8	VOLUME4[15:8]
140152D4	ABIST_CTL0	8	ABIST_CTL_L
140152D8	ABIST_CTL1	8	ABIST_CTL_H
14015304	DUMMY	8	Registar Dummy
14015310	CRC_CTRL	8	CRC_CTRL
14015314	CRC_RESULT_L	8	CRC_RESULT_L
14015318	CRC_RESULT_H	8	CRC_RESULT_H
1401531C	COLOR_DEPTH_CTRL_L	8	COLOR_DEPTH_CTRL_L
14015320	COLOR_DEPTH_CTRL_M	8	COLOR_DEPTH_CTRL_M
14015324	COLOR_DEPTH_CTRL_H	8	COLOR_DEPTH_CTRL_H
14015340	CFG_REG_CRC0	8	CRC_CTRL
14015344	CFG_REG_CRC1	8	CHECK_NUM
14015348	CFG_REG_CRC2	8	GOLDEN_SUM_L
1401534C	CFG_REG_CRC3	8	GOLDEN_SUM_H
14015350	CFG_REG_CRC4	8	CHECKSUM_MON_H
14015354	CFG_REG_CRC5	8	CHECKSUM_MON_L
14015380	VIDEO_CFG_0	8	VIDEO_CFG_0
14015384	VIDEO_CFG_	8	VIDEO_CFG_1

	1		
14015388	VIDEO_CFG_2	8	VIDEO_CFG_2
1401538C	VIDEO_CFG_3	8	VIDEO_CFG_3
14015390	VIDEO_CFG_4	8	VIDEO_CFG_4

14015000 **VEN_IDL** **Vendor ID** **00**

Bit	7	6	5	4	3	2	1	0
Mne	VEN_IDL							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VEN_IDL	This register contains vendor ID low 8 bit.

14015004 **VEN_IDH** **Vendor ID** **00**

Bit	7	6	5	4	3	2	1	0
Mne	VEN_IDH							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VEN_IDH	This register contains vendor ID high 8 bit.

14015008 **DEV_IDL** **Device ID** **00**

Bit	7	6	5	4	3	2	1	0
Mne	VEN_IDL							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VEN_IDL	This register contains device ID low 8 bit.

1401500C **DEV_IDH** **Device ID** **00**

Bit	7	6	5	4	3	2	1	0

Mne	VEN_IDH							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VEN_IDH	**This register contains device ID high 8 bit.**

14015010 **DEV_REV** **Revision ID** **00**

Bit	7	6	5	4	3	2	1	0
Mne	DEV_REV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		DEV_REV	**This register contains the revision ID.**

14015014 **INT** **Interrupt** **00**

Bit	7	6	5	4	3	2	1	0
Mne	Ctrl_Pkt_Done	NCTS_Done	INF_Done	IFM_ERR	FIFO_U	FIFO_O	HDCP	MDI
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		Ctrl_Pkt_Done	1: An interrupt is generated at end of General Control package transmission.
6		NCTS_Done	1: An interrupt is generated at end of each NCTS package transmission.
5		INF_Done	**Suggested to mask this interrupt in the auto-repeat transmission for InfoFrame.** 1: An interrupt is generated at end of each InfoFrame package transmission.
4		IFM_ERR	**Resend for the InfoFrame bytes is suggested.** 1: This bit is set when InfoFrame is received with error via I2C.
3		FIFO_U	1: when no more audio sample received in the FIFO for two consecutive video lines.
2		FIFO_O	1: when FIFO error occurs, such as overflow
1		HDCP	**HDCP interrupt** 1: Interrupt from HDCP (AKSV error/correct, V_Rdy, Ri_Rdy, BKSV_Err)
0		MDI	**Monitor Detect Interrupt** 1: Either PORD or HPLG detection signal has changed logic level.

14015018 **INT_MASK** **Int mask** **00**

Bit	7	6	5	4	3	2	1	0
Mne	Ctrl_Pkt_Done	NCTS_Done	INF_Done	IFM_ERR	FIFO_U	FIFO_O	HDCP	MDI
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		Ctrl_Pkt_Done	**Set 1 to mask the interrupt of Ctrl_Pkt_Done**
6		NCTS_Done	**Set 1 to mask the interrupt of NCTS_DONE**
5		INF_Done	**Set 1 to mask the interrupt of INF_Done**
4		IFM_ERR	**Set 1 to mask the interrupt of IFM_ERR**
3		FIFO_U	**Set 1 to mask the interrupt of FIFO_U**
2		FIFO_O	**Set 1 to mask the interrupt of FIFO_O**
1		HDCP	**All interrupt relate to HDCP(AKSV error/correct, V_Rdy, Ri_Rdy, BKSV_Err) in HDCP_STA are msaked**
0		MDI	**Set 1 to mask the interrupt of MDI**

1401501C **CTRL** **CTRL** **00**

Bit	7	6	5	4	3	2	1	0
Mne	AVMUTE	AVI_En	Audio_En	MPEG_En	SPD_En	Generic_En	HDCP_Debug	SRST
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		AVMUTE	**Host setting this bit to ONE activates general control package transmission with 'Set_AVMUTE'in the next vertical blanking interval. Host clearing this bit (changing from ONE to ZERO) results in general control package transmission with 'Clear_AVMUTE' in the next vertical blanking interval.** Note: While the matrix is uploaded with any InfoFrame, data in the matrix uploaded will be discarded if micro-controller is trying to write to the register file for update. If VSYNC comes in the middle of uploading, Other package or NULL packet will be transmitted instead of the InfoFrame.
6		AVI_En	1: AVI InfoFrame packets will be automatically transmitted once per blanking interval; 0: InfoFrame packet will be transmitted only when host updates the InfoFrame, i.e., microcontroller writes to InfoFrame register.
5		Audio_En	1: Audio InfoFrame packets will be automatically transmitted once per blanking interval; 0: InfoFrame packet will be transmitted only when host updates the InfoFrame, i.e., microcontroller writes to InfoFrame register.
4		MPEG_En	1: MPEG InfoFrame packets will be automatically transmitted once per blanking interval; 0: InfoFrame packet will be transmitted only when host updates the

Bit(s)	Mnemonic	Name	Description
			InfoFrame, i.e., microcontroller writes to InfoFrame register.
3		SPD_En	1: SPD InfoFrame packets will be automatically transmitted once per blanking interval; 0: InfoFrame packet will be transmitted only when host updates the InfoFrame, i.e., microcontroller writes to InfoFrame register.
2		Generic_En	1: Any other InfoFrame packets stored in generic buffer will be automatically transmitted once in every blanking interval; 0: InfoFrame packet will be transmitted only when host updates the InfoFrame, i.e., microcontroller writes to InfoFrame register.
1		HDCP_Debug	**This register is used for HDCP debug.** 1: if AN[63:0] are written from micro-controller and want to monitor Mi[63:0] from 0x54~0x70
0		SRST	**It is active HIGH.** 1: reset the audio and video FIFO control logic, period control state machine.

14015020 **STATUS** **Status** **00**

Bit	7	6	5	4	3	2	1	0
Mne	AVI_Done	Audio_Done	MPEG_Done	SPD_Done	Generic_Done	Audio_Sample	PORD	HTPLG
Type	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		AVI_Done	**This flag is set when transmission for AVI InfoFrame is done.**
6		Audio_Done	**This flag is set when transmission for Audio InfoFrame is done.**
5		MPEG_Done	**This flag is set when transmission for MPEG InfoFrame is done.**
4		SPD_Done	**This flag is set when transmission for SPD InfoFrame is done.**
3		Generic_Done	**This flag is set when transmission for other InfoFrame is done.**
2		Audio_Sample	**ONE indicates that transmission for audio sample is OK. This status is held for one video frame to allow host access. It is cleared by VSYNC and set at audio-sample package completion.**
1		PORD	**Powered-On Receiver Detect status bit. It is reflection of the output from the power-on receiver detection circuit. Any level change shall result in interrupt.**
0		HTPLG	**Hot Plug Detect status bit reflects the state of HTPLG pin. Any level change shall result in interrupt.**

14015024 **CFG0** **Config register 0** **00**

Bit	7	6	5	4	3	2	1	0
Mne	FRQ_Rate		W_Length		EDG_I2S	LRCK	I2S_Mode	
Type	RW		RW		RW	RW	RW	
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:6		FRQ_Rate	**FRQ_Rate[1:0], BCLK/(2*LRCK)** 00: 32 01: 24 10: 16 11: -
5:4		W_Length	**W_Length[1:0], Word Length** 00: 24 01: 16 10: Reserved 11: Reserved
3		EDG_I2S	**Clock edge select for I2S bus data latch.** 1: Falling Edge; 0: Rising Edge.
2		LRCK	**{LRCK, I2S_Mode[1:0]}, Remark** 000: Left Justified, Left channel is sent at LRCK level HIGH 100: Left Justified, Left channel is sent at LRCK level LOW 001: Right Justified, Left channel is sent at LRCK level HIGH 101: Right Justified, Left channel is sent at LRCK level LOW 010: I2S, Left channel is sent at LRCK level LOW 110: I2S, Left channel is sent at LRCK level HIGH x11
1:0		I2S_Mode	

14015028 **CFG1** **Config register 1** **0A**

Bit	7	6	5	4	3	2	1	0
Mne	DST_ECO_2	DST_ECO_1	Non_Zero_CTS	CTLo_value	HDCP_Debug	DVI	SPDIF	EDG_Sel
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
7		DST_ECO_2	**CTS counter use N/128 for dst mode option**
6		DST_ECO_1	**CTS counter use N/64 for dst mode option**
5		Non_Zero_CTS	1: Set CTS[19:0] = 20'h0 if the previous computed CTS[19:0] is same as old; 0: Set CTS[19:0] non-zero if the previous computed CTS[19:0] is same as old .
4		CTLo_value	1: Set CTLo be '0' in control period; 0: Set CTLo be '1' in control period .
3		HDCP_Debug	1: Mi/AN[63:0] will be updated in 0x15~0x1C (AN) read back register ; 0: Mi/AN[63:0] will be fixed in 0x15~0x1C (AN) read back register.
2		DVI	**The transmitter works in DVI Mode if this bit is set HIGH.**

Bit(s)	Mnemonic	Name	Description
			1: DVI mode
1		SPDIF	**When this bit is set HIGH, the SPDIF channel is active for audio interface instead of I2S.**
0		EDG_Sel	**Clock edge selection for data latching for SPDIF interface.** 1: data should be latched on falling edge of the clock; 0: rising edge of the clock.

1401502C **CFG2** **Config register 2** **03**

Bit	7	6	5	4	3	2	1	0
Name	ACLK_INV	Notice_EN				Reserved		
Type	RW	RW				RW		
Reset	0	0				0	1	1

Bit(s)	Mnemonic	Name	Description
7		ACLK_INV	1: Using inverse audio clock to latch data for audio down sample mode from 192k to 48k in I2S.
6		Notice_EN	**To avoid audio packet jitter problem.** 1: Notice enable will always be '1'; 0: Notice enable will always be '1' only in vertical blank period (Default).

14015030 **CFG3** **Config register 3** **40**

Bit	7	6	5	4	3	2	1	0
Name	KSV_load_start	CTL_sel	AES_KEY_INDEX					
Type	RW	RW	RW					
Reset	0	1	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		KSV_load_start	**Start load KSV form external E2PROM or MCM E2PROM.**
6		CTL_sel	1: Select CTL0 ~ CTL3 no delay; 0: Select CTL0 ~ CTL3 delay 1T.
5:0		AES_KEY_INDEX	**Select input AES_KEY_INDEX[5:0] from 0~39, total 40 of 56 bits HDCP key.**

14015034 **CFG4** **Config register 4** **25**

Bit	7	6	5	4	3	2	1	0
Name		Clear_mute_en	Mute_clear_sel	AES_KEY_LOAD	DST_ECO_3	Frame_128_sel	DST_ECO_0	HDMI_bypass_analog
Type		RW	RW	RW	RW	RW	RW	RW
Reset		0	1	0	0	1	0	1

Bit(s)	Mnemonic	Name	Description
6		Clear_mute_en	**If use new clear mute function, set '1' to send clear mute packet.**
5		Mute_clear_sel	1: to select original clear mute function. 0: new clear mute function, can clear mute without set mute.
4		AES_KEY_LOAD	1: to set a load pulse and get the 56 bits HDCP key in the AES_KEY_INDEX[5:0].
3		DST_ECO_3	**For modify the bug of dst in the spec** 1: normal; 0: dst_frame will delay 8T
2		Frame_128_sel	
1		DST_ECO_0	**For modify the bug of dst in the spec** 1: dst_frame invalid ,no sample data in the dst mode; 0: will sample data always in the dst mode
0		HDMI_bypass_analog	**Will use the bit with inverter** 1: normal; 0: clk can get without analog pll.

14015038 **CFG5** **Config register 5** **30**

Bit	7	6	5	4	3	2	1	0
Name		CD_Ratio_Spdif			NCTS_ECO_2			
Type		RW			RW			
Reset		0	1	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
6:4		CD_Ratio_Spdif	**Clock-Data Ratio for SPDIF interface configuration** CD_Ratio[2:0], Clock:Data, Remark 000: x 001: 1/1, 128*fs 010: 2/1, 256*fs 011: 3/1, 384*fs (Default) 100: 4/1, 512*fs 101: 5/1, 640*fs 110: 6/1, 768*fs 111: 7/1, 896*fs
3:0		NCTS_ECO_2	**2ACR solution** [3]: ncts_program_error [2]: fetch data if ncts_ready assert [1]: fetch data if ncts_ready_en_sync assert [0]: ncts_ready no assert if state starts 0x0 is turn on

14015040 **WR_BKSV0** **Write for BKSV[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BKSV							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BKSV	Byte 0 is the LSB, byte 4 is MSB. The byte 5 should be written last and then trigger the authentication process in this transmitter.

14015044 **WR_BKSV1** **Write for BKSV[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BKSV							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BKSV	Byte 0 is the LSB, byte 4 is MSB. The byte 5 should be written last and then trigger the authentication process in this transmitter.

14015048 **WR_BKSV2** **Write for BKSV[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BKSV							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BKSV	Byte 0 is the LSB, byte 4 is MSB. The byte 5 should be written last and then trigger the authentication process in this transmitter.

1401504C **WR_BKSV3** **Write for BKSV[31:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BKSV							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BKSV	**Byte 0 is the LSB, byte 4 is MSB. The byte 5 should be written last and then trigger the authentication process in this transmitter.**

14015050 **WR_BKSV4** **Write for BKSV[39:32]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BKSV							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BKSV	**Byte 0 is the LSB, byte 4 is MSB. The byte 5 should be written last and then trigger the authentication process in this transmitter.**

14015054 **WR_AN0** **Write for AN[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	**The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.**

14015058 **WR_AN1** **Write for AN[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

1401505C **WR_AN2** **Write for AN[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

14015060 **WR_AN3** **Write for AN[31:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

14015064 **WR_AN4** **Write for AN[39:32]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

14015068 **WR_AN5** **Write for AN[47:40]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

1401506C **WR_AN6** **Write for AN[55:48]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be

Bit(s)	Mnemonic	Name	Description
			generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

14015070 **WR_AN7** **Write for AN[63:56]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AN							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AN	The value can be read by firmware and used in the authentication process. Alternatively, this value can be generated by firmware or hardware (board), and then written into this register. Eight Bytes: Byte 0 is the LSB, Byte 7 is the MSB.

14015074 **RD_AKSV0** **Read for AKSV[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AKSV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AKSV	In none Host_Key mode, the AKSV is stored in EPPROM and read by internal I2C master every time when Load_Key bit is set. After power-on, KSV is automatically loaded from the EPPROM via I2C master. Once it is verified by hardware with AKSV_Rdy interrupt bit set, microcontroller reads AKSV and then writes into receiver to initiate the authentication process. AKSV is registered as shown in the table above to allow access by host. Byte 0 is the LSB, byte 4 is MSB. The least bit of the Byte 0 is the least bit of the AKSV.

14015078 **RD_AKSV1** **Read for AKSV[**

15:8]

Bit	7	6	5	4	3	2	1	0
Name	AKSV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AKSV	In none Host_Key mode, the AKSV is stored in EPPROM and read by internal I2C master every time when Load_Key bit is set. After power-on, KSV is automatically loaded from the EPPROM via I2C master. Once it is verified by hardware with AKSV_Rdy interrupt bit set, microcontroller reads AKSV and then writes into receiver to initiate the authentication process. AKSV is registered as shown in the table above to allow access by host. Byte 0 is the LSB, byte 4 is MSB. The least bit of the Byte 0 is the least bit of the AKSV.

1401507C **RD_AKSV2** **Read for AKSV[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AKSV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AKSV	In none Host_Key mode, the AKSV is stored in EPPROM and read by internal I2C master every time when Load_Key bit is set. After power-on, KSV is automatically loaded from the EPPROM via I2C master. Once it is verified by hardware with AKSV_Rdy interrupt bit set, microcontroller reads AKSV and then writes into receiver to initiate the authentication process. AKSV is registered as shown in the table above to allow access by host. Byte 0 is the LSB, byte 4 is MSB. The least bit of the Byte 0 is the least bit of the AKSV.

14015080 **RD_AKSV3** **Read for AKSV[31:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name	AKSV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AKSV	**In none Host_Key mode, the AKSV is stored in EPPROM and read by internal I2C master every time when Load_Key bit is set. After power-on, KSV is automatically loaded from the EPPROM via I2C master. Once it is verified by hardware with AKSV_Rdy interrupt bit set, microcontroller reads AKSV and then writes into receiver to initiate the authentication process. AKSV is registered as shown in the table above to allow access by host. Byte 0is the LSB, byte 4 is MSB. The least bit of the Byte 0 is the least bit of the AKSV.**

14015084 RD_AKSV4 Read for AKSV[39:32] 00

Bit	7	6	5	4	3	2	1	0
Name	AKSV							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		AKSV	**In none Host_Key mode, the AKSV is stored in EPPROM and read by internal I2C master every time when Load_Key bit is set. After power-on, KSV is automatically loaded from the EPPROM via I2C master. Once it is verified by hardware with AKSV_Rdy interrupt bit set, microcontroller reads AKSV and then writes into receiver to initiate the authentication process. AKSV is registered as shown in the table above to allow access by host. Byte 0is the LSB, byte 4 is MSB. The least bit of the Byte 0 is the least bit of the AKSV.**

14015088 RI_0 RI[7:0] 00

Bit	7	6	5	4	3	2	1	0
Name	Ri							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		Ri	**It can be read by the host and compared against the Ri value read from the receiver to sure that the encryption process on the transmitter and receiver are synchronized.**

1401508C RI_1 RI[15:8] 00

Bit	7	6	5	4	3	2	1	0
Name	Ri							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		Ri	**It can be read by the host and compared against the Ri value read from the receiver to sure that the encryption process on the transmitter and receiver are synchronized.**

14015090 **KEY_PORT** **Key port** **00**

Bit	7	6	5	4	3	2	1	0
Name	Key_port							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		Key_port	**The key contains 40 words and each word is of 56 bits. The key is written into the port in the sequence as shown in the Key Map below.** Note: 40-byte keys must be written continuously. Key Map Index Content 0x0000 Key00[0] 0x0001 Key00[1] 0x0002 Key00[2] 0x0003 Key00[3] 0x0004 Key00[4] 0x0005 Key00[5] 0x0006 Key00[6] 0x0007 Key01[0] 0x0008 Key01[1] ... ... 0x0140 Key39[7] Note: Key00[0] contains the lowest 8 bits of the first words. Its least significant bit is the least significant bit of the first word.

14015094 **KSVKIST** **KSV lisy** **00**

Bit	7	6	5	4	3	2	1	0
Name	KSV_list							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		KSV_list	**This port is used by host to supply KSV list to transmitter to compute the KSV list integrity verification value for HDCP repeater authentication.**

14015098 **I2CM_ADDR** **I2C Addr** **B0**

Bit	7	6	5	4	3	2	1	0
Name	I2CM_ADDR							
Type	RW							
Reset	1	0	1	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		I2CM_ADDR	This register allows host to program I2C device ADDR8 of the EEPROM that contains the KSV and key set for HDCP authentication process. The default value after Power_On reset is defined from the PAD_GPO_3 and PAD_GPO_4 as following:

1401509C **BIST0** **BIST[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BIST							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BIST	This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.

140150A0 **BIST1** **BIST[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BIST							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BIST	This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.

140150A4 **BIST2** **BIST[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	BIST							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		BIST	This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.

140150A8 **BIST3** **BIST[29:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name			BIST					
Type			RW					
Reset			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
5:0		BIST	This register provides 30 bit initial value for the LFSR used as Build-In-Self-Test (BIST) for TMDS buffer test.

140150B8 **HDCP_STA** **HDCP status** **00**

Bit	7	6	5	4	3	2	1	0
Name			AKSV_Rdy	V_Rdy	V_Match	RI_Rdy	BKSV_Err	ENC_On
Type			RO	RO	RO	RO	RO	RO
Reset			0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
5		AKSV_Rdy	AKSV is automatically loaded by hardware after power-on reset if DIS_AUTO bit is set LOW. This interrupt register status bit is set once it is verified by hardware. Assertion of this bit shall trigger the interrupt to host microcontroller. If the KSV is detected invalid, interrupt shall also be generated but without any status bit set. This bit is cleared by host writing ONE or by power-on reset or CP_Rst.
4		V_Rdy	Interrupt register status bit; ONE indicates that the KSV list integrity verification value is ready. Assertion of this bit shall trigger the interrupt to host. This bit can be cleared by host writing ONE to this bit or by power-on reset or CP_Rst.
3		V_Match	This status bit is set by hardware when the computed V is equal to the V' value read from repeater. This bit is cleared by power-on reset or CP_Rst.
2		RI_Rdy	Interrupt register status bit; ONE indicates that Ri value is available. Assertion of this bit shall trigger the interrupt to the microcontroller. This bit is cleared by host writing ONE or by power-on reset or CP_Rst. This bit is also set by hardware every 128 frames after the authentication is successfully completed. It is cleared by host writing ONE or by power-on reset or CP_Rst. Note: This interrupt should be serviced with highest priority so that RI value can be read within 100ms.

Bit(s)	Mnemonic	Name	Description
1		BKSV_Err	**Interrupt register status bit. ONE indicates that the received BKSV value from host microcontroller does not have the proper structure (containing 20 zeros and 20 ones). Assertion of this bit shall trigger the interrupt to the microcontroller. This bit is cleared when a new BKSV value is written, or by host writing ONE to it, or by power-on reset or CP_Rst.**
0		ENC_On	**This status bit indicates that the encryption is enabled and encryption is in process. It cleared by power-on reset or CP_Rst.**

140150BC **HDCP_CTRL** **HDCP control** **00**

Bit	7	6	5	4	3	2	1	0
Name	SHA_En	Host_Key		RX_Rptr	AN_Stop	CP_Rstb	Authen_En	ENC_En
Type	RW	RW		RW	RW	RW	RW	RW
Reset	0	0		0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		SHA_En	**After the KSV list and its integrity verification value V' are written into the transmitter's registers, host shall set this bit to enable the second stage of authentication if RX_Rptr bit is set.**
6		Host_Key	**The key will be loaded by host via I2C register.** 1: I2C master read from external E2PROM or MCM will be disabled.
4		RX_Rptr	**This bit is set HIGH by firmware prior to authentication if the receiver is a repeater.**
3		AN_Stop	**When this bit is cleared, the cipher engine is allowed to free-run and the WR_AN registers will cycle through a sequence of pseudo-random values. When this bit is set, cycling stops and the value is held in the WR_AN register and can be read by firmware to initiate the AN register in the receiver.** Note: This bit is automatically cleared under any of the following conditions: 1. Power-on Reset 2. BKSV_Err 3. RX_Rptr is changed 4. Cipher Reset: CP_Rst
2		CP_Rstb	**This bit is set HIGH to reset the cipher engine, active HIGH. It must be cleared to start normal HDCP operation. Cipher reset is recommended after aborted authentication and before re-attempt for new authentication. In the host-key mode, this bit must be set HIGH to reset the cipher before key load.**
1		Authen_En	**Set this bit to start authentication process.**
0		ENC_En	**Encryption Enable. When this bit is set, the transmitter encrypts all data.**

140150C0 **V0** **Repeater Hash V[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150C4 **V1** **Repeater Hash V[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150C8 **V2** **Repeater Hash V[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150CC **V3** **Repeater Hash V[31:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150D0 **V4** **Repeater Hash V[39:32]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150D4 **V5** **Repeater Hash V[47:40]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150D8 **V6** **Repeater Hash V[55:48]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150DC **V7** **Repeater Hash V[63:56]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150E0 **V8** **Repeater Hash V[71:64]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150E4 **V9** **Repeater Hash V[79:72]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150E8 **V10** **Repeater Hash V[87:80]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150EC **V11** **Repeater Hash V[95:88]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.**

Bit(s)	Mnemonic	Name	Description
			Byte 0 is the LSB, byte 19 is MSB.

140150F0 **V12** **Repeater Hash V[103:96]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150F4 **V13** **Repeater Hash V[111:104]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150F8 **V14** **Repeater Hash V[119:112]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

140150FC **V15** **Repeater Hash V[127:120]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

14015100 **V16** **Repeater Hash V[135:128]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

14015104 **V17** **Repeater Hash V[143:136]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

14015108 **V18** **Repeater Hash V[151:144]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

1401510C **V19** **Repeater Hash V[159:152]** **00**

Bit	7	6	5	4	3	2	1	0
Name	V							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		V	**Note:Multi-byte write with auto-increment shall be supported for this port.** Byte 0 is the LSB, byte 19 is MSB.

14015154 **I2S_UV** **I2S_UV** **3C**

Bit	7	6	5	4	3	2	1	0
Name	Normal_Info_Inv	TMDS_Debug	CH_En				U	V
Type	RW	RW	RW				RW	RW
Reset	0	0	1	1	1	1	0	0

Bit(s)	Mnemonic	Name	Description
7		Normal_Info_Inv	**HDMI1.1 packet delivering using generic infoframe when set to .** 1: default; 0: for HDMI 1.0 generic infoframe delivering.
6		TMDS_Debug	1: Let HDMI three channels output = { lfsr2[9:0], lfsr1[9:0], lfsr0[9:0] }; 0: Normal operation (Default).
5:2		CH_En	**I2S channel enable. When any of the bits is set HIGH, the corresponding channel as illustrated in table below is enabled.**
1		U	**User bit.**
0		V	**Validity bit for audio sample supplied via I2S channel.**

14015158 **CTRL_1** **Info autosend ctrl** **00**

Bit	7	6	5	4	3	2	1	0
Name				GAMUT_EN	ISRC2_EN	ISRC1_EN	ACP_EN	VS_EN
Type				RW	RW	RW	RW	RW
Reset				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4		GAMUT_EN	**GAMUT packet auto send.**
3		ISRC2_EN	**ISRC2 packet auto send.**
2		ISRC1_EN	**ISRC1 packet auto send.**
1		ACP_EN	**ACP packet auto send.**
0		VS_EN	**Vendor-Specific infoframe auto send.**

1401515C **STATUS_1** **Info status** **00**

Bit	7	6	5	4	3	2	1	0
Name				GAMUT_DONE	ISRC2_DONE	ISRC1_DONE	ACP_DONE	VS_DONE
Type				RO	RO	RO	RO	RO
Reset				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
4		GAMUT_DONE	**GAMUT packet was send.**
3		ISRC2_DONE	**ISRC2 packet was send.**
2		ISRC1_DONE	**ISRC1 packet was send.**
1		ACP_DONE	**ACP packet was send.**
0		VS_DONE	**Vendor-Specific infoframe was send.**

14015160 **CTS_CTRL** **Soft control NCTS** **00**

Bit	7	6	5	4	3	2	1	0
Name								SOFT_NCTS
Type								RW
Reset								0

Bit(s)	Mnemonic	Name	Description
0		SOFT_NCTS	1: NCTS packet will take the CTS value from register calculated by software; 0: Hardware generated CTS value will used for NCTS instead.

14015164 **CTS0** **CTS[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	CTS0							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		CTS0	**Host microcontroller may check the CTS value via these registers.**

14015168 **CTS1** **CTS[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	CTS1							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		CTS1	**Host microcontroller may check the CTS value via these registers.**

1401516C **CTS2** **CTS[19:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name					CTS2			
Type					RW			
Reset					0	0	0	0

Bit(s)	Mnemonic	Name	Description
3:0		CTS2	**Host microcontroller may check the CTS value via these registers.**

14015178 **DIV_RESET** **00**

Bit	7	6	5	4	3	2	1	0
Name								DIV_RESET
Type								RW
Reset								0

Bit(s)	Mnemonic	Name	Description
0		DIV_RESET	0: normal 1: U/V swap

1401517C **AUDIO_CFG** **01**

Bit	7	6	5	4	3	2	1	0
Name	DSD_SEL	LR_MIX	LR_INV	DSD_INV	DST_NORMAL_DOUBLE	SACD_DST	HIGH_BIT_RATE	AUDIO_ZERO
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
7		DSD_SEL	**HDMI 1.2 SACD DSD audio packets enable.**
6		LR_MIX	**HDMI 1.2 SACD for MTK solution. (2ch DSD data mix at one pin)**
5		LR_INV	**HDMI 1.2 SACD for MTK solution L/R inverse or not**
4		DSD_INV	**HDMI 1.2 SACD DSD data inverse or not.**
3		DST_NORMAL_DOUBLE	**HDMI 1.3 SACD DST packet normal/double**
2		SACD_DST	**HDMI 1.3 SACD DST audio packets enable**
1		HIGH_BIT_RATE	**HDMI 1.3 high bit rate stream (DTS HD, Dolby true HD)**
0		AUDIO_ZERO	**HDMI output audio mute pattern.**

14015180 **SRAM_DELSEL** **SRAM delay sel** **AA**

Bit	7	6	5	4	3	2	1	0
Name	HDCP_SRAM_DELSEL		AUDIO_SRAM_DELSEL		IIR_SRAM_DELSEL1		IIR_SRAM_DELSEL0	
Type	RW		RW		RW		RW	
Reset	1	0	1	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
7:6		HDCP_SRAM_DELSEL	**Sram delay sel**
5:4		AUDIO_SRAM_DELSEL	**Sram delay sel**
3:2		IIR_SRAM_DELSEL1	**Sram delay sel**
1:0		IIR_SRAM_DELSEL0	**Sram delay sel**

14015184 **NCTS** **Write for N,CTS** **00**

Bit	7	6	5	4	3	2	1	0
Name	NCTS_PORT							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		NCTS_PORT	**The bytes must be written in the sequence as shown in table below. When the last byte is written, a ready signal, NCTS_Rdy, should be generated to indicate the readiness of the NCTS data for matrix uploading. It is cleared when host writes this port.** Byte, {7 6 5 4 3 2 1 0} 0, {0x00} 1, {4'b0000, CTS[19:16]} 2, CTS[15:8] 3, CTS[7:0] 4, {4'b0000, N[19:16]} 5, N[15:8] 6, N[7:0]

14015188 **IFM_PORT** **Information Port** **00**

Bit	7	6	5	4	3	2	1	0
Name	IFM_PORT							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		IFM_PORT	**Similar to N_CTS registers, these bytes must be written into the register file in the sequence shown in the table below. The number of bytes, i.e., the InfoFrame length is defined in its header. Check-sum byte is calculated by software in host. A ready signal, InfoFrm_Rdy, is**

Bit(s)	Mnemonic	Name	Description
			generated when the last byte is written. Byte, {7 6 5 4 3 2 1 0} 0, Checksum 1~N, Data Byte 1 ~ Data Byte N

1401518C **CH_SWITCH0** **CH_SWITCH[7:0]** **88**

Bit	7	6	5	4	3	2	1	0
Name	SD2_SW_L		SD1_SW			SD0_SW		
Type	RW		RW			RW		
Reset	1	0	0	0	1	0	0	0

Bit(s)	Mnemonic	Name	Description
7:6		SD2_SW_L	**HDMI 3rd channel data from ASDATA[SD2_SW]** {0x190[0], 0x18C[7:6]} 0: SD0_SW 1: SD1_SW 2: SD2_SW (default) 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
5:3		SD1_SW	**HDMI 2nd channel data from ASDATA[SD1_SW]** 0: SD0_SW 1: SD1_SW (default) 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
2:0		SD0_SW	**HDMI 1st channel data from ASDATA[SD0_SW]** 0: SD0_SW (default) 1: SD1_SW 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW

14015190 **CH_SWITCH1** **CH_SWITCH[15:8]** **36**

Bit	7	6	5	4	3	2	1	0

Name	SD5_SW_L	SD4_SW			SD3_SW			SD2_SW_H
Type	RW	RW			RW			RW
Reset	0	0	1	1	0	1	1	0

Bit(s)	Mnemonic	Name	Description
7		SD5_SW_L	**HDMI 6th channel data from ASDATA[SD5_SW]** {0x194[1:0], 0x190[7]} 0: SD0_SW 1: SD1_SW 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
6:4		SD4_SW	**HDMI 5th channel data from ASDATA[SD4_SW]** 0: SD0_SW 1: SD1_SW 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
3:1		SD3_SW	**HDMI 4th channel data from ASDATA[SD3_SW]** 0: SD0_SW 1: SD1_SW 2: SD2_SW 3: SD3_SW (default) 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
0		SD2_SW_H	***Please see 0x18C[7:6]**

14015194 **CH_SWITCH2** **CH_SWITCH[23:16]** **16**

Bit	7	6	5	4	3	2	1	0
Name	SD7_SW			SD6_SW			SD5_SW_H	
Type	RW			RW			RW	
Reset	0	0	0	1	0	1	1	0

Bit(s)	Mnemonic	Name	Description
7:5		SD7_SW	**HDMI 8th channel data from ASDATA[SD7_SW]** 0: SD0_SW 1: SD1_SW 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW

Bit(s)	Mnemonic	Name	Description
			6: SD6_SW 7: SD7_SW **HDMI 7th channel data from ASDATA[SD6_SW]**
4:2		SD6_SW	0: SD0_SW 1: SD1_SW 2: SD2_SW 3: SD3_SW 4: SD4_SW 5: SD5_SW 6: SD6_SW 7: SD7_SW
1:0		SD5_SW_H	***Please see 0x190[7]**

14015198 **LR_SWITCH** **LR_SWITCH** **00**

Bit	7	6	5	4	3	2	1	0
Name	C_SD3_LR	C_SD2_LR	C_SD1_LR	C_SD0_LR	SD3_LR	SD2_LR	SD1_LR	SD0_LR
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		C_SD3_LR	**HDMI 4th channel L/R status inverse or not.**
6		C_SD2_LR	**HDMI 3rd channel L/R status inverse or not.**
5		C_SD1_LR	**HDMI 2nd channel L/R status inverse or not.**
4		C_SD0_LR	**HDMI 1st channel L/R status inverse or not.**
3		SD3_LR	**HDMI 4th channel L/R inverse or not.**
2		SD2_LR	**HDMI 3rd channel L/R inverse or not.**
1		SD1_LR	**HDMI 2nd channel L/R inverse or not.**
0		SD0_LR	**HDMI 1st channel L/R inverse or not.**

1401519C **InfoFrame_Header_1** **InfoFrame version** **00**

Bit	7	6	5	4	3	2	1	0
Name	InfoFrame_version							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		InfoFrame_version	**ADDR8, {7 6 5 4 3 2 1 0}** 0x19C, InfoFrame_Version 0x1A0, InfoFrame_Type

Bit(s)	Mnemonic	Name	Description
			0x1A4, InfoFrame_Length Programming Sequence: 0x1A0 -> 0x19C -> 0X1A4 (0X1A0 will assert busy) Note: ? Prior to the InfoFrame bytes write, three InfoFrame Header bytes must be written. ? The check sum is calculated by hardware as well as host. The check-sum includes the three byte header. Hardware verifies the check sum by comparing it with the received. If mismatch, set a status bit and interrupt microcontroller for re-send. Host is required to write the correct number of bytes as indicated by InfoFrame_Length provided in the InfoFrame Header. ? Audio, AVI, MPEG, SPD and other InfoFrame are buffered separately by five register files. Auto repeat-transmission can be enabled by setting the enable bits defined in CTL register respectively. Each InfoFrame will be transmitted once in every Vertical Blanking Interval in auto repeat mode. The arbitration for transmission from the five buffers is Audio, AVI, MPEG and followed by SPD and then the genaric. Any InfoFrame content update will result in transmission for the modified InfoFrame in the next immediate Data Island

140151A0 **InfoFrame_Header_2** **InfoFrame type** **00**

Bit	7	6	5	4	3	2	1	0
Name	InfoFrame_type							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		InfoFrame_type	

140151A4 **InfoFrame_Header_3** **InfoFrame length** **00**

Bit	7	6	5	4	3	2	1	0
Name	InfoFrame_length							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		InfoFrame_length	

140151B0 **Shift_length_R2** **Shift_length_R2** **00**

Bit	7	6	5	4	3	2	1	0
Name	Audio_20bit	Audio_mute						
Type	RW	RW						
Reset	0	0						

Bit(s)	Mnemonic	Name	Description
7		Audio_20bit	**Audio_20bit 20 bit audio sample mode**
6		Audio_mute	1: Stop sending audio packet to HDMI output; 0: Normal operation(Default).

140151B4 **Mix_Ctrl** **Mix_Ctrl** **12**

Bit	7	6	5	4	3	2	1	0
Name	Disable_flat	Lrck_cycle_sel		Set_u_zero	C_bit_sel	Down_sample	BYPASS_VOLUME	Src_en
Type	RW	RW		RW	RW	RW	RW	RW
Reset	0	0	0	1	0	0	1	0

Bit(s)	Mnemonic	Name	Description
7		Disable_flat	**Disable HDMI flat bit for transmit SPDIF raw data.**
6:5		Lrck_cycle_sel	00: 64 cycles(SPDIF), 01: 32 cycles(I2S), 10: 24 cycles(I2S), 11: 16 cycles(I2S).
4		Set_u_zero	**SRC set U bit to zero.**
3		C_bit_sel	**SRC select C bit from left channel (0) or right channel (1).**
2		Down_sample	0: 2x down sample, 1: 4x down sample
1		BYPASS_VOLUME	**Bypass volume control hardware**
0		Src_en	**Sampling rate converter enable**

140151B8 **IIR_Filter** **IIR_Filter** **00**

Bit	7	6	5	4	3	2	1	0
Name	Filter_SRAM							
Type	WO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		Filter_SRAM	**IIR filter coefficients (192*8).**

140151BC **Shift_length_Lo** **Shift_length_Lo** **00**

Bit	7	6	5	4	3	2	1	0
Name	Shift_length_Left_L							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		Shift_length_Left_L	**CTL[3:0] source control for DVI mode.** If Shift_length_Left[4] == 1'b1, CTL[3:0] = I2Sdata[3:0] pins. If Shift_length_Left[4] == 1'b0, CTL[3:0] = Shift_length_Left[3:0].

140151C0 **Shift_length_L1** **Shift_length_L1** **00**

Bit	7	6	5	4	3	2	1	0
Name	Error_mask			Src_4N	Src_2N	Shift_length_Left_H		
Type	RW			RW	RW	RW		
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:5		Error_mask	**Mask the error in iir_filter** bit7:mask U bit error bit6:mask C bit error bit5:mask LRCK cycle error
4		Src_4N	**CTS counter use 4N for 4X down sample**
3		Src_2N	**CTS counter use2N for 2X down sample**
2:0		Shift_length_Left_H	

140151C4 **AOUT_CFG** **00**

Bit	7	6	5	4	3	2	1	0
Name	Burst_Preamble_En	FIFO_ADAP_CTRL	SOFT_RST_HDMI_FIFO	SPDIF_READY_RST			AOUT_BNUM_SEL	
Type	RW	RW	RW	RW			RW	
Reset	0	0	0	0			0	0

Bit(s)	Mnemonic	Name	Description
7		Burst_Preamble_En	**SPDIF frame start when bust preamble**
6		FIFO_ADAP_CTRL	**Constrain that Pa and Pb align with subpacket 0 in HBR packet**
5		SOFT_RST_HDMI_FIFO	**Audio fifo soft reset**
4		SPDIF_READY_RST	**Reset SPDIF ready information such as burst preamble.**
1:0		AOUT_BNUM_SEL	**Audio output bit number selection** 0x: 24bit 10: 20bit 11: 16bit

14015200 L_STATUS_0 L_STATUS[7:0] 00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_0							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_0	**Left channel status[7:0].**

14015204 L_STATUS_1 L_STATUS[15:8] 00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_1							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_1	**Left channel status[15:8].**

14015208 L_STATUS_2 L_STATUS[23:16] 00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_2							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_2	**Left channel status[23:16].**

1401520C **L_STATUS_3** **L_STATUS[31:23]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_3							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_3	**Left channel status[31:24].**

14015210 **L_STATUS_4** **L_STATUS[39:32]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_4							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_4	**Left channel status[39:32].**

14015214 **L_STATUS_5** **L_STATUS[47:40]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_5							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_5	**Left channel status[47:40].**

14015218 **L_STATUS_6** **L_STATUS6[55:48]** **00**

Bit	7	6	5	4	3	2	1	0

Name	L_STATUS_6							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_6	**Left channel status[55:48].**

1401521C **L_STATUS_7** **L_STATUS[63:56]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_7							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_7	**Left channel status[63:56].**

14015220 **L_STATUS_8** **L_STATUS[71:64]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_8							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_8	**Left channel status[71:64].**

14015224 **L_STATUS_9** **L_STATUS[79:72]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_9							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_9	**Left channel status[79:72].**

14015228 **L_STATUS_10** **L_STATUS[87:80]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_10							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_10	**Left channel status[87:80].**

1401522C **L_STATUS_11** **L_STATUS1[95:88]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_11							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_11	**Left channel status[95:88].**

14015230 **L_STATUS_12** **L_STATUS[103:96]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_12							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_12	**Left channel status[103:96].**

14015234 **L_STATUS_13** **L_STATUS[111:104]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_13							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_13	**Left channel status[111:104].**

14015238 **L_STATUS_14** **L_STATUS[119:112]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_14							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_14	**Left channel status[119:112].**

1401523C **L_STATUS_15** **L_STATUS[127:120]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_15							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_15	**Left channel status[127:120].**

14015240 **L_STATUS_16** **L_STATUS[135:128]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_16							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_16	**Left channel status[135:128].**

14015244 **L_STATUS_17** **L_STATUS[143:136]** **00**

Bit	7	6	5	4	3	2	1	0

Name	L_STATUS_17							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_17	**Left channel status[143:136].**

14015248 **L_STATUS_18** **L_STATUS[151:144]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_18							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_18	**Left channel status[151:144].**

1401524C **L_STATUS_19** **L_STATUS[159:152]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_19							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_19	**Left channel status[159:152].**

14015250 **L_STATUS_20** **L_STATUS[167:160]** **00**

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_20							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_20	**Left channel status[167:160].**

14015254	L_STATUS_21	L_STATUS[175:168]	00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_21							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_21	**Left channel status[175:168].**

14015258	L_STATUS_22	L_STATUS[183:176]	00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_22							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_22	**Left channel status[183:176].**

1401525C	L_STATUS_23	L_STATUS[191:184]	00

Bit	7	6	5	4	3	2	1	0
Name	L_STATUS_23							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		L_STATUS_23	**Left channel status[191:184].**

14015260	R_STATUS_0	R_STATUS[7:0]	00

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_0							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_0	**Right channel status[7:0].**

14015264 **R_STATUS_1** **R_STATUS[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_1							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_1	**Right channel status[15:8].**

14015268 **R_STATUS_2** **R_STATUS[23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_2							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_2	**Right channel status[23:16].**

1401526C **R_STATUS_3** **R_STATUS[31:24]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_3							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_3	**Right channel status[31:24].**

14015270 **R_STATUS_4** **R_STATUS[39:32]** **00**

Bit	7	6	5	4	3	2	1	0

Name	R_STATUS_4							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_4	**Right channel status[39:32].**

14015274 **R_STATUS_5** **R_STATUS[47:40]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_5							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_5	**Right channel status[47:40].**

14015278 **R_STATUS_6** **R_STATUS[55:48]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_6							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_6	**Right channel status[55:48].**

1401527C **R_STATUS_7** **R_STATUS[63:56]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_7							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_7	**Right channel status[63:56].**

14015280	R_STATUS_8	R_STATUS[71:64]	00

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_8							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_8	**Right channel status[71:64].**

14015284	R_STATUS_9	R_STATUS[79:72]	00

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_9							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_9	**Right channel status[79:72].**

14015288	R_STATUS_10	R_STATUS[87:80]	00

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_10							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_10	**Right channel status[87:80].**

1401528C	R_STATUS_11	R_STATUS[95:88]	00

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_11							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_11	**Right channel status[95:88].**

14015290 **R_STATUS_12** **R_STATUS[103:96]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_12							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_12	**Right channel status[103:96].**

14015294 **R_STATUS_13** **R_STATUS[111:104]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_13							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_13	**Right channel status[111:104].**

14015298 **R_STATUS_14** **R_STATUS[119:112]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_14							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_14	**Right channel status[119:112].**

1401529C **R_STATUS_15** **R_STATUS[127:120]** **00**

Bit	7	6	5	4	3	2	1	0

Name	R_STATUS_15							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_15	**Right channel status[127:120].**

140152A0 **R_STATUS_16** **R_STATUS[135:128]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_16							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_16	**Right channel status[135:128].**

140152A4 **R_STATUS_17** **R_STATUS[143:126]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_17							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_17	**Right channel status[143:136].**

140152A8 **R_STATUS_18** **R_STATUS[151:144]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_18							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_18	**Right channel status[151:144].**

140152AC **R_STATUS_19** **R_STATUS[159:152]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_19							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_19	**Right channel status[159:152].**

140152B0 **R_STATUS_20** **R_STATUS[167:160]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_20							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_20	**Right channel status[167:160].**

140152B4 **R_STATUS_21** **R_STATUS[175:168]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_21							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_21	**Right channel status[175:168].**

140152B8 **R_STATUS_22** **R_STATUS[183:176]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_22							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_22	**Right channel status[183:176].**

140152BC **R_STATUS_23** **R_STATUS[191:184]** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_STATUS_23							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_STATUS_23	**Right channel status[191:184].**

140152C0 **VOLUME0** **VOLUME_TARGET[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	VOLUME_TARGET0							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VOLUME_TARGET0	**Volume target [7:0].**

140152C4 **VOLUME1** **VOLUME_TARGET[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	VOLUME_TARGET1							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VOLUME_TARGET1	**Volume target [15:8].**

140152C8 **VOLUME2** **VOLUME_TARGET [23:16]** **00**

Bit	7	6	5	4	3	2	1	0
Name	VOLUME_TARGET2							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VOLUME_TARGET2	**Volume target [23:16].**

140152CC **VOLUME3** **VOLUME_STEP[7:0]** **00**

Bit	7	6	5	4	3	2	1	0
Name	VOLUME_STEP0							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VOLUME_STEP0	**Volume step [7:0].**

140152D0 **VOLUME4** **VOLUME4[15:8]** **00**

Bit	7	6	5	4	3	2	1	0
Name	VOLUME_STEP1							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		VOLUME_STEP1	**Volume step [15:8].**

140152D4 **ABIST_CTL0** **ABIST_CTL_L** **0A**

Bit	7	6	5	4	3	2	1	0
Name	Vsync_polarity	Hsync_polarity	Video_format					

Type	RW	RW	RW					
Reset	0	0	0	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
7		Vsync_polarity	**ABIST mode video output Vsync polarity** 0: Low polarity (Default). 1: High polarity.
6		Hsync_polarity	**ABIST mode video output Hsync polarity** 0: Low polarity (Default). 1: High polarity.
5:0		Video_format	**ABIST mode video output timing** 000010: 720x480p @ 60Hz. 000011: 1280x720p @ 60HZ. 000100: 1920x1080i @ 60 Hz. 000101: 1440x480i @ 60 Hz. 001001: 1440x480p @ 60 Hz. 001010: 1920x1080p @ 60 Hz(Default). 001011: 720x576p @ 50 Hz. 001010: 1280x720p @ 50 Hz. 001011: 1920x1080i @ 50 Hz. 001110: 1440x576i @ 50 Hz. 010010: 1440x576p @ 50 Hz. 010011: 1920x1080p @ 50 Hz.

140152D8 **ABIST_CTL1** **ABIST_CTL_H** **00**

Bit	7	6	5	4	3	2	1	0
Name	ABIST_EN				Speed_mode	Data_format		
Type	RW				RW	RW		
Reset	0				0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		ABIST_EN	**Enable ABIST mode, video data and timing will gen from HDMI internal.** 1: enable 0: disable
3		Speed_mode	**Output frame with few line number.**
2:0		Data_format	**ABIST mode video output data, {R[15:0],G[15:0],B[15:0]}** 000: stair output. 001: {aaaaaaaaaaaa}. 010: {555555555555}. 011: {111111111111}.

14015304 **DUMMY** **Register Dummy** **90**

Bit	7	6	5	4	3	2	1	0
Name	chn2_sel		chn1_sel		chn0_sel			new_gcp_ctrl
Type	RW		RW		RW			RW
Reset	1	0	0	1	0	0		0

Bit(s)	Mnemonic	Name	Description
7:6		chn2_sel	2'b00: chn2 output is data[9:0] 2'b01: chn2 output is data[19:10] 2'b10: chn2 output is data[29:20]
5:4		chn1_sel	2'b00: chn2 output is data[9:0] 2'b01: chn2 output is data[19:10] 2'b10: chn2 output is data[29:20]
3:2		chn0_sel	2'b00: chn2 output is data[9:0] 2'b01: chn2 output is data[19:10] 2'b10: chn2 output is data[29:20]
0		new_gcp_ctrl	1: Use new gcp packet send merge deepcolor with avmute 0: Use orig way send gcp packet

14015310 **CRC_CTRL** **CRC_CTRL** **00**

Bit	7	6	5	4	3	2	1	0
Name							clr_crc_result	init_crc
Type							WO	WO
Reset							0	0

Bit(s)	Mnemonic	Name	Description
1		clr_crc_result	1: clear CRC reslut
0		init_crc	1: Start CRC

14015314 **CRC_RESULT_L** **CRC_RESULT_L** **00**

Bit	7	6	5	4	3	2	1	0
Name	crc_result_l							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		crc_result_l	

14015318 **CRC_RESULT_H** **CRC_RESULT** **00**

_H

Bit	7	6	5	4	3	2	1	0
Name	crc_result_h							
Type	RO							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		crc_result_h	

1401531C **COLOR_DEPTH_CTRL_L** **COLOR_DEPTH_CTRL_L** **00**

Bit	7	6	5	4	3	2	1	0
Name	V_data_lpf_sign_new	V_data_lpf_sign	U_data_lpf_sign_new	U_data_lpf_sign	Y_data_lpf_sign_new	Y_data_lpf_sign	lpf_enable	lpf_sel
Type	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		V_data_lpf_sign_new	1: V channel lpf signed with new way
6		V_data_lpf_sign	**V data lpf sign** 1: V channel lpf signed
5		U_data_lpf_sign_new	1: U channel lpf signed with new way
4		U_data_lpf_sign	**U data lpf sign** 1: U channel lpf signed
3		Y_data_lpf_sign_new	1: Y channel lpf signed with new way
2		Y_data_lpf_sign	**Y data lpf sign** 1: Y channel lpf signed
1		lpf_enable	1: enable 5taps lpf 0: disable
0		lpf_sel	1: select 5taps output as video data 0: lsb fill with msb or zero

14015320 **COLOR_DEPTH_CTRL_M** **COLOR_DEPTH_CTRL_M** **00**

Bit	7	6	5	4	3	2	1	0
Name	lpf_threshold_lsb				sync_timing_delay			lsb_fill_with_msb
Type	RW				RW			RW

Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4		lpf_threshold_lsb	
3:1		sync_timing_delay	**hsync,vysnc,de delay xxT**
0		lsb_fill_with_msb	1: lsb fill with msb 0: lsd fill with zero

14015324 **COLOR_DEPTH_CTRL_H** **COLOR_DEPTH_CTRL_H** **08**

Bit	7	6	5	4	3	2	1	0
Name	lpf_threshold_msb							
Type	RW							
Reset	0	0	0	0	1	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		lpf_threshold_msb	

14015340 **CFG_REG_CRC0** **CRC_CTRL** **00**

Bit	7	6	5	4	3	2	1	0
Name	check_done	err_flag				crc_ncts_info_sel	crc_clear	crc_enable
Type	RO	RO				RW	RW	RW
Reset	0	0				0	0	0

Bit(s)	Mnemonic	Name	Description
7		check_done	1: check done 0: check not done
6		err_flag	1: CRC error 0: crc ok
2		crc_ncts_info_sel	1: ncts infoframe packet crc check 0: i2s/spdif crc check
1		crc_clear	
0		crc_enable	1: audio crc enable 0: disable

14015344 **CFG_REG_CRC1** **CHECK_NU** **00**

M

Bit	7	6	5	4	3	2	1	0
Name	check_num							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		check_num	

14015348 **CFG_REG_CRC2** **GOLDEN_SUM_L** **00**

Bit	7	6	5	4	3	2	1	0
Name	GOLDEN_SUM_L							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		GOLDEN_SUM_L	

1401534C **CFG_REG_CRC3** **GOLDEN_SUM_H** **00**

Bit	7	6	5	4	3	2	1	0
Name	GOLDEN_SUM_H							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		GOLDEN_SUM_H	

14015350 **CFG_REG_CRC4** **CHECKSUM_MON_H** **00**

Bit	7	6	5	4	3	2	1	0
Name	CHECKSUM_MON_H							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		CHECKSUM_MON_H	

14015354 **CFG_REG_CRC5** **CHECKSUM_MON_L** **00**

Bit	7	6	5	4	3	2	1	0
Name	CHECKSUM_MON_L							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		CHECKSUM_MON_L	

14015380 **VIDEO_CFG_0** **VIDEO_CFG_0** **00**

Bit	7	6	5	4	3	2	1	0
Name	B_VALUE_L							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		B_VALUE_L	

14015384 **VIDEO_CFG_1** **VIDEO_CFG_1** **00**

Bit	7	6	5	4	3	2	1	0
Name	G_VALUE_L				B_VALUE_H			
Type	RW				RW			
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:4		G_VALUE_L	
3:0		B_VALUE_H	

14015388 **VIDEO_CFG_2** **VIDEO_CFG_2** **00**

Bit	7	6	5	4	3	2	1	0
Name	G_VALUE_H							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		G_VALUE_H	

1401538C **VIDEO_CFG_3** **VIDEO_CFG_3** **00**

Bit	7	6	5	4	3	2	1	0
Name	R_VALUE_L							
Type	RW							
Reset	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		R_VALUE_L	

14015390 **VIDEO_CFG_4** **VIDEO_CFG_4** **00**

Bit	7	6	5	4	3	2	1	0
Name	VIDEO_SOURCE_SEL				R_VALUE_H			
Type	RW				RW			
Reset	0				0	0	0	0

Bit(s)	Mnemonic	Name	Description
7		VIDEO_SOURCE_SEL	1: video data is from normal path 0: video data is from RGB_VALUE from register config
3:0		R_VALUE_H	

17 MIPI TX

17.1 Introduction

The MIPI TX module, as well as MIPI_TX_Config described in the following sections, is used to control MIPI TX related registers for MIPI DPHY macro. This analog macro includes design of SDM PLL, bandgap, LDO core, lane control, GPI pads, output enable, etc.

17.2 Feature list

- Bandgap control
- LDO core power and configuration
- SDM PLL configuration
- SSC control
- Analog function related settings and status
- Output pads control and electronic features
- GPI pads control
- Software-control mode for each lanes
- Lane swap for clock and data lanes

17.3 Register Definition

Module name: MIPI_TX0_CONFIG Base address: (+10010000h)

Address	Name	Width	Register Function
10010000	DSI0_CON	32	DSI Configuration Register
10010004	DSI0_CLOCK_LANE	32	DSI Clock Lane Configuration Register
10010008	DSI0_DATA_LANE_0	32	DSI Data Lane 0 Configuration Register
1001000C	DSI0_DATA_LANE_1	32	DSI Data Lane 1 Configuration Register
10010010	DSI0_DATA_LANE_2	32	DSI Data Lane 2 Configuration Register
10010014	DSI0_DATA_LANE_3	32	DSI Data Lane 3 Configuration Register
10010040	DSI_TOP_CON	32	DSI Top Configuration Register
10010044	DSI_BG_CON	32	DSI BG Configuration Register
10010050	DSI_PLL_CON0	32	DSI PLL Configuration 0 Register
10010054	DSI_PLL_CON	32	DSI PLL Configuration 1 Register

	1		
10010058	DSI_PLL_CON2	32	DSI PLL Configuration 2 Register
1001005C	DSI_PLL_CON3	32	DSI PLL Configuration 3 Register
10010060	DSI_PLL_CHG	32	DSI PLL Charge Register
10010064	DSI_PLL_TOP	32	DSI PLL TOP Configuration Register
10010068	DSI_PLL_PWR	32	DSI PLL Power Control Register
10010070	DSI_RGS	32	DSI RGS Register
10010074	DSI_GPIO_EN	32	DSI GPIO Enable Register
1001007C	DSI_PHY_SEL	32	DSI MIPI TX PHY Lane Swap Selection
10010080	DSI_SW_CTRL_EN	32	DSI Software Control Enable Register
10010084	DSI_SW_CTRL_CON0	32	DSI Software Control Configuration Register 0
10010088	DSI_SW_CTRL_CON1	32	DSI Software Control Configuration Register 1
1001008C	DSI_SW_CTRL_CON2	32	DSI Software Control Configuration Register 2
10010090	DSI_DBG_CON	32	DSI Debug Control Register

10010000 DSI0_CON DSI Configuration Register 00000400

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		RG_DSI0_LPRXCD_SEL			RG_DSI0_LPTX_CLMP_EN	RG_DSI0_DSICLK_FREQ_SEL	RG_DSI0_PHYCLK_SEL			RG_DSI0_LD_IDX_SEL			RG_DSI0_BCLK_SEL		RG_DSI0_CKG_LDOOUT_EN	RG_DSI0_LDOCORE_EN
Type		RW			RW	RW	RW			RW			RW		RW	RW
Reset		0	0	0	0	1	0	0		0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
14:12		RG_DSI0_LPRXCD_SEL	**RG_DSI0_LPRXCD_SEL:** 3'd0: LANEC 3'd1: LANE0 3'd2: LANE1 3'd3: LANE2 3'd4: LANE3
11		RG_DSI0_LPTX	**DSI0_LPTX_CLMP_EN:**

Bit(s)	Mnemonic	Name	Description
		_CLMP_EN	1'b0: dc clamp diable 1'b1: dc clamp enable
10		RG_DSIo_DSICLK_FREQ_SEL	**DSIo DSICLK frequency select:** 1'b0: 2X of PHYCLK 1'b1: 1X of PHYCLK
9:8		RG_DSIo_PHYCLK_SEL	**DSIo PHYCLK phase select:** 2'b00: align with BCLK 2'b01: align with BCLKQ 2'b10: align with BCLKB 2'b11: align with BCLKQB
6:4		RG_DSIo_LD_IDX_SEL	**DSIo LD_IDX phase select:** 3'b000: 1/4T delay from CKQ 3'b001: 3/8T delay from CKQ etc.
3:2		RG_DSIo_BCLK_SEL	**DSIo byteclk phase select:** 2'b00: align with CK 2'b01: align with CKQ 2'b10: align with CKB 2'b11: align with CKQB
1		RG_DSIo_CKG_LDOOUT_EN	**DSIo CKG LDO output Enable:** 1'b0: disable 1'b1: enable
0		RG_DSIo_LDOCORE_EN	**DSIo LDO core Enable:** 1'b0: disable 1'b1: enable

10010004 **DSIo_CLOCK_LANE** **DSI Clock Lane Configuration Register** **00001820**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				**RG_DSIo_LNTC_CKLANE_EN**	**RG_DSIo_LNTC_RT_CODE**						**RG_DSIo_LNTC_PHI_SEL**	**RG_DSIo_LNTC_LPTX_IMINUS**	**RG_DSIo_LNTC_LPTX_IPLUS2**	**RG_DSIo_LNTC_LPTX_IPLUS1**	**RG_DSIo_LNTC_LOOPBACK_EN**	**RG_DSIo_LNTC_LDOOUT_EN**
Type				RW	RW						RW	RW	RW	RW	RW	RW
Reset				1	1	0	0	0			1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
12		RG_DSIo_LNTC_CKLANE_EN	**RG_DSIo_LNTC_CKLANE_EN** 1'b1: enable

Bit(s)	Mnemonic	Name	Description
			1'b0: disable
11:8		RG_DSIo_LNTC_RT_CODE	**DSIo CK Lane HS impedance control code:** 4'b0000: maximum impedance 4'b1111: minimum impedance
5		RG_DSIo_LNTC_PHI_SEL	**DSIo CK Lane clock phase select:** 1'b0: first output bit of clock lane is 0 1'b1: first output bit of clock lane is 1
4		RG_DSIo_LNTC_LPTX_IMINUS	**DSIo CK Lane LP driver current adjust:** 1'b0: normal 1'b1: -10%
3		RG_DSIo_LNTC_LPTX_IPLUS2	**DSIo CK Lane LP driver N current adjust:** 1'b0: normal 1'b1: +10%
2		RG_DSIo_LNTC_LPTX_IPLUS1	**DSIo CK Lane LP driver P current adjust:** 1'b00: normal 1'b01: +10%
1		RG_DSIo_LNTC_LOOPBACK_EN	**DSIo CK Lane Loopback enable:** 1'b0: diable 1'b1: enable
0		RG_DSIo_LNTC_LDOOUT_EN	**DSIo CK lane LDO output Enable:** 1'b0: disable 1'b1: enable

10010008 **DSIo_DATA_LANE_0** **DSI Data Lane 0 Configuration Register** **00000400**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					**RG_DSIo_LNTo_CKLANE_EN**	**RG_DSIo_LNTo_RT_CODE**				**RG_DSIo_LNTo_LPCD_IMINUS**	**RG_DSIo_LNTo_LPCD_IPLUS**	**RG_DSIo_LNTo_LPTX_IMINUS**	**RG_DSIo_LNTo_LPTX_IPLUS2**	**RG_DSIo_LNTo_LPTX_IPLUS1**	**RG_DSIo_LNTo_LOOPBACK_EN**	**RG_DSIo_LNTo_LDOOUT_EN**
Type					RW	RW				RW	RW	RW	RW	RW	RW	RW
Reset					0	1	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
11		RG_DSIo_LNTo_CKLANE_EN	**RG_DSIo_LNTo_CKLANE_EN** 1'b1: enable 1'b0: disable
10:7		RG_DSIo_LNTo	**DSIo Laneo HS impedance control code:**

Bit(s)	Mnemonic	Name	Description
		_RT_CODE	4'b0000: maximum impedance 4'b1111: minimum impedance
6		RG_DSIo_LNTo_LPCD_IMINUS	**DSIo Laneo LP driver current adjust:** 1'b0: normal 1'b1: -10%
5		RG_DSIo_LNTo_LPCD_IPLUS	**DSIo Laneo LP driver N current adjust:** 1'b0: normal 1'b1: +10%
4		RG_DSIo_LNTo_LPTX_IMINUS	**DSIo Laneo LP driver current adjust:** 1'b0: normal 1'b1: -10%
3		RG_DSIo_LNTo_LPTX_IPLUS2	**DSIo Laneo LP driver N current adjust:** 1'b0: normal 1'b1: +10%
2		RG_DSIo_LNTo_LPTX_IPLUS1	**DSIo Laneo LP driver P current adjust:** 1'b00: normal 1'b01: +10%
1		RG_DSIo_LNTo_LOOPBACK_EN	**DSIo Laneo Loopback enable:** 1'b0: diable 1'b1: enable
0		RG_DSIo_LNTo_LDOOUT_EN	**DSIo laneo LDO output Enable:** 1'b0: disable 1'b1: enable

1001000C **DSIo_DATA_LANE_1** **DSI Data Lane 1 Configuration Register** **00000100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							**RG_DSIo_LNT1_CKLANE_EN**	**RG_DSIo_LNT1_RT_CODE**				**RG_DSIo_LNT1_LPTX_IMINUS**	**RG_DSIo_LNT1_LPTX_IPLUS2**	**RG_DSIo_LNT1_LPTX_IPLUS1**	**RG_DSIo_LNT1_LOOPBACK_EN**	**RG_DSIo_LNT1_LDOOUT_EN**
Type							RW	RW				RW	RW	RW	RW	RW
Reset							0	1	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
9		RG_DSIo_LNT1_CKLANE_EN	**RG_DSIo_LNT1_CKLANE_EN** 1'b1: enable 1'b0: disable
8:5		RG_DSIo_LNT1	**DSIo Lane1 HS impedance control code:**

Bit(s)	Mnemonic	Name	Description
		_RT_CODE	4'b0000: maximum impedance 4'b1111: minimum impedance
4		RG_DSIo_LNT1_LPTX_IMINUS	**DSIo Lane1 LP driver current adjust:** 1'b0: normal 1'b1: -10%
3		RG_DSIo_LNT1_LPTX_IPLUS2	**DSIo Lane1 LP driver N current adjust:** 1'b0: normal 1'b1: +10%
2		RG_DSIo_LNT1_LPTX_IPLUS1	**DSIo Lane1 LP driver P current adjust:** 1'b00: normal 1'b01: +10%
1		RG_DSIo_LNT1_LOOPBACK_EN	**DSIo Lane1 Loopback enable:** 1'b0: diable 1'b1: enable
0		RG_DSIo_LNT1_LDOOUT_EN	**DSIo lane1 LDO output Enable:** 1'b0: disable 1'b1: enable

10010010 **DSIo_DATA_LANE_2** **DSI Data Lane 2 Configuration Register** **00000100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							RG_DSIo_LNT2_CKLANE_EN	RG_DSIo_LNT2_RT_CODE				RG_DSIo_LNT2_LPTX_IMINUS	RG_DSIo_LNT2_LPTX_IPLUS2	RG_DSIo_LNT2_LPTX_IPLUS1	RG_DSIo_LNT2_LOOPBACK_EN	RG_DSIo_LNT2_LDOOUT_EN
Type							RW	RW				RW	RW	RW	RW	RW
Reset							0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bit(s)	Mnemonic	Name	Description
9		RG_DSIo_LNT2_CKLANE_EN	**RG_DSIo_LNT2_CKLANE_EN** 1'b1: enable 1'b0: disable
8:5		RG_DSIo_LNT2_RT_CODE	**DSIo lane2 HS impedance control code:** 4'b0000: maximum impedance 4'b1111: minimum impedance
4		RG_DSIo_LNT2_LPTX_IMINUS	**DSIo lane2 LP driver current adjust:** 1'b0: normal 1'b1: -10%

Bit(s)	Mnemonic	Name	Description
3		RG_DSIo_LNT2_LPTX_IPLUS2	**DSIo lane2 LP driver N current adjust:** 1'bo: normal 1'b1: +10%
2		RG_DSIo_LNT2_LPTX_IPLUS1	**DSIo lane2 LP driver P current adjust:** 1'boo: normal 1'bo1: +10%
1		RG_DSIo_LNT2_LOOPBACK_EN	**DSIo lane2 Loopback enable:** 1'bo: diable 1'b1: enable
0		RG_DSIo_LNT2_LDOOUT_EN	**DSIo lane2 LDO output Enable:** 1'bo: disable 1'b1: enable

10010014 **DSIo_DATA_LANE_3** **DSI Data Lane 3 Configuration Register** **00000100**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name							**RG_DSIo_LNT3_CKLANE_EN**	**RG_DSIo_LNT3_RT_CODE**				**RG_DSIo_LNT3_LPTX_IMINUS**	**RG_DSIo_LNT3_LPTX_IPLUS2**	**RG_DSIo_LNT3_LPTX_IPLUS1**	**RG_DSIo_LNT3_LOOPBACK_EN**	**RG_DSIo_LNT3_LDOOUT_EN**
Type							RW	RW				RW	RW	RW	RW	RW
Reset							0	1	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
9		RG_DSIo_LNT3_CKLANE_EN	**RG_DSIo_LNT3_CKLANE_EN** 1'b1: enable 1'bo: disable
8:5		RG_DSIo_LNT3_RT_CODE	**DSIo lane3 HS impedance control code:** 4'bo000: maximum impedance 4'b1111: minimum impedance
4		RG_DSIo_LNT3_LPTX_IMINUS	**DSIo lane3 LP driver current adjust:** 1'bo: normal 1'b1: -10%
3		RG_DSIo_LNT3_LPTX_IPLUS2	**DSIo lane3 LP driver N current adjust:** 1'bo: normal 1'b1: +10%
2		RG_DSIo_LNT3_LPTX_IPLUS1	**DSIo lane3 LP driver P current adjust:** 1'boo: normal

Bit(s)	Mnemonic	Name	Description
			1'b01: +10%
1		RG_DSI0_LNT3_LOOPBACK_EN	**DSI0 lane3 Loopback enable:** 1'b0: diable 1'b1: enable
0		RG_DSI0_LNT3_LDOOUT_EN	**DSI0 lane3 LDO output Enable:** 1'b0: disable 1'b1: enable

10010040 **DSI_TOP_CON** **DSI Top Configuration Register** **00000080**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**RG_DSI_PRESERVE**			**RG_DSI_DEBUG_INPUT_EN**	**RG_DSI_PAD_TIE_LOW_EN**	**RG_DSI_LNT_AIO_SEL**			**RG_DSI_LNT_IMP_CAL_CODE**				**RG_DSI_LNT_TESTMODE_EN**	**RG_DSI_LNT_IMP_CAL_EN**	**RG_DSI_LNT_HS_BIAS_EN**	**RG_DSI_LNT_INTR_EN**
Type	RW			RW	RW	RW			RW				RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:13		RG_DSI_PRESERVE	**DSI Preserved registers**
12		RG_DSI_DEBUG_INPUT_EN	**Enable the debug input to CKG** 1'b0: normal operation 1'b1: disable CKG clock, enable the debug signals
11		RG_DSI_PAD_TIE_LOW_EN	**DSI 8 pads tie-low enable:** 1'b0: floating or depends on line driver. 1'b1: tie to AVSS18 ground
10:8		RG_DSI_LNT_AIO_SEL	**DSI analog debug mode output select:** 3'b000: output VREF 0.2V 3'b001: output VREF 0.32V 3'b010: output VREF 0.4V 3'b011: output VREF 0.72V 3'b100: output VREF 0.86V 3'b101: output VREF 1.2V 3'b110: output VCCK09 3'b111: output PLL_TST
7:4		RG_DSI_LNT_IMP_CAL_CODE	**DSI HS impedance calibration code:** 4'b0000: maximum impedance 4'b1111: minimum impedance
3		RG_DSI_LNT_T	**DSI analog debug mode enable:**

Bit(s)	Mnemonic	Name	Description
		ESTMODE_EN	1'b0: disable 1'b1: enable
2		RG_DSI_LNT_IMP_CAL_EN	**DSI HS impedance calibration enable:** 1'b0: disable 1'b1: enable
1		RG_DSI_LNT_HS_BIAS_EN	**DSI HS bias enable:** 1'b0: disable 1'b1: enable
0		RG_DSI_LNT_INTR_EN	**DSI HS bias internal R enable:** 1'b0: use external R 1'b1: use internal R

10010044 DSI_BG_CON DSI BG Configuration Register 88492480

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DSI_BG_R2_TRIM				RG_DSI_BG_R1_TRIM					RG_DSI_V02_SEL			RG_DSI_V032_SEL			RG_DSI_V04_SEL[2:2]
Type	RW				RW					RW			RW			RW
Reset	1	0	0	0	1	0	0	0		1	0	0	1	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DSI_V04_SEL[1:0]		RG_DSI_V072_SEL		RG_DSI_V10_SEL				RG_DSI_V12_SEL			RG_DSI_BG_FAST_CHARGE	RG_DSI_BG_DIV		RG_DSI_BG_CKEN	RG_DSI_BG_CORE_EN
Type	RW		RW		RW				RW			RW	RW		RW	RW
Reset	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:28		RG_DSI_BG_R2_TRIM	**DSI BGR R2 trim:** ORed with RG_CSI_BG_R2_TRIM[3:0] 4'b0000: maximum resistance 4'b1000: typical resistance 4'b1111: minimum resistance
27:24		RG_DSI_BG_R1_TRIM	**DSI BGR R1 trim:** ORed with RG_CSI_BG_R1_TRIM[3:0] 4'b0000: maximum resistance 4'b1000: typical resistance 4'b1111: minimum resistance
22:20		RG_DSI_V02_SEL	**Select 0.2V output voltage:** 3'b000: minimum voltage 3'b100: typical voltage

Bit(s)	Mnemonic	Name	Description
			3'b111: maximum voltage
19:17		RG_DSI_V032_SEL	**Select 0.32V output voltage:** 3'b000: minimum voltage 3'b100: typical voltage 3'b111: maximum voltage
16:14		RG_DSI_V04_SEL	**Select 0.4V output voltage:** 3'b000: minimum voltage 3'b100: typical voltage 3'b111: maximum voltage
13:11		RG_DSI_V072_SEL	**Select 0.72V output voltage:** 3'b000: minimum voltage 3'b100: typical voltage 3'b111: maximum voltage
10:8		RG_DSI_V10_SEL	**Select AVDD10 output voltage:** 3'b000: minimum voltage 3'b100: typical voltage 3'b111: maximum voltage
7:5		RG_DSI_V12_SEL	**Select 1.2V output voltage:** 3'b0000: minimum voltage 3'b0100: typical voltage 3'b0111: maximum voltage
4		RG_DSI_BG_FAST_CHARGE	**BG bias low-pass filter fast charge enable** 1'b0: Disable, settling time 30us 1'b1: Enable, settling time 1us
3:2		RG_DSI_BG_DIV	**BG chopper clock divisor control:** 2'b00: /2 2'b01: /4 2'b10: /8 2'b11: /16
1		RG_DSI_BG_CKEN	**BG chopper clock enable** 1'b0: Disable 1'b1: Enable
0		RG_DSI_BG_CORE_EN	**BG Enable** OR with RG_CSI_BG_CORE_EN 1'b0: Power off 1'b1: Enable

10010050 **DSI_PLL_CON0** **DSI PLL Configuration 0 Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				RG_DSIo_MPPLL_VOD	RG_DSIo_MPPLL_MON	RG_DSIo_MPPLL_MON	RG_DSIo_MPPLL_POSDIV			RG_DSIo_MPPLL_TXDIV1		RG_DSIo_MPPLL_TXDIVo		RG_DSIo_MPPLL_PREDIV		RG_DSIo_MPPLL_PLL

10010050 **DSI_PLL_CON0** **DSI PLL Configuration 0 Register** **00000000**

				EN	REF_EN	VCO_EN										EN
Type				RW	RW	RW	RW			RW		RW		RW		RW
Reset				0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
12		RG_DSI0_MPPLL_VOD_EN	**MPPLL VOD EN** 1'b0: AVDD10 for PLL is NOT over-drived 10% to 1.15V for Vcore = 1.15V 1'b1: AVDD10 for PLL IS over-drived 10% to 1.15V for Vcore = 1.05V
11		RG_DSI0_MPPLL_MONREF_EN	**Monitor reference Enable** 1'b0: Disable 1'b1: Enable
10		RG_DSI0_MPPLL_MONVC_EN	**Monitor Vctrl Enable** 1'b0: Disable 1'b1: Enable
9:7		RG_DSI0_MPPLL_POSDIV	**PLL Post divide ratio** 3'b000: /1 3'b001: /2 3'b010: /4 3'b011: /8 3'b100: /16 Others: Clock Gating
6:5		RG_DSI0_MPPLL_TXDIV1	**PLL0 VCO Post-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4
4:3		RG_DSI0_MPPLL_TXDIV0	**PLL0 VCO Post-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4
2:1		RG_DSI0_MPPLL_PREDIV	**Pre-divider ratio** 2'b00: /1 2'b01: /2 2'b1X: /4
0		RG_DSI0_MPPLL_PLL_EN	**PLL Enable** 1'b0: Disable 1'b1: Enable

10010054 **DSI_PLL_CON1** **DSI PLL Configuration 1 Register** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DSI0_MPPLL_SDM_SSC_PRD															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														RG_D	RG_D	RG_D

10010054 **DSI_PLL_CON1** **DSI PLL Configuration 1 Register** **00000002**

														SIo_MPPLL_SDM_SSC_EN	SIo_MPPLL_SDM_SSC_PH_INIT	SIo_MPPLL_SDM_FRA_EN
Type														RW	RW	RW
Reset														0	1	0

Bit(s)	Mnemonic	Name	Description
31:16		RG_DSIo_MPPLL_SDM_SSC_PRD	**PLL SDM SSC period** 16'do: Min 16'd65536: Max Counter triggered by reference 26MHz clock
2		RG_DSIo_MPPLL_SDM_SSC_EN	**PLL SDM SSC enable** 1'bo: Disable 1'b1: Enable
1		RG_DSIo_MPPLL_SDM_SSC_PH_INIT	**PLL SDM SSC phase initial** 1'bo: Upward 1'b1: Downward
0		RG_DSIo_MPPLL_SDM_FRA_EN	**PLL SDM Fractional Mode En** 1'bo: Integer Mode 1'b1: Fractional Mode

10010058 **DSI_PLL_CON2** **DSI PLL Configuration 2 Register** **50000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name		RG_DSIo_MPPLL_SDM_PCW[30:16]														
Type		RW														
Reset		1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DSIo_MPPLL_SDM_PCW[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
30:0		RG_DSIo_MPPLL_SDM_PCW	**PLL Feedback divide ratio** 7bit integer + 24bit fractional

1001005C **DSI_PLL_CON3** **DSI PLL Configuration 3 Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RG_DSIo_MPPLL_SDM_SSC_DELTA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_DSIo_MPPLL_SDM_SSC_DELTA1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		RG_DSIo_MPPLL_SDM_SSC_DELTA	**PLL SDM SSC amplitude** 16'do: Min 16'd65536: Max
15:0		RG_DSIo_MPPLL_SDM_SSC_DELTA1	**PLL SDM SSC amplitude 1** 16'do: Min 16'd65536: Max

10010060 **DSI_PLL_CHG** **DSI PLL Charge Register** **00000001**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																RG_DSIo_MPPLL_SDM_PCW_CHG
Type																RW
Reset																1

Bit(s)	Mnemonic	Name	Description
0		RG_DSIo_MPPLL_SDM_PCW_CHG	**PLL SDM Feedback divide ratio update signal** update divide ratio at 0 to 1 falling edge. Go back to 1 after the registers are updated.

10010064 **DSI_PLL_TOP** **DSI PLL TOP Configuration Register** **00000000**

10010064 **DSI_PLL_TOP** **DSI PLL TOP Configuration Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RG_MPPLL_PRESERVE												RG_MPPLL_TSTSEL		RG_MPPLL_TSTCK_EN	RG_MPPLL_TST_EN
Type	RW												RW		RW	RW
Reset	0	0	0	0	0	0	0	0					0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8		RG_MPPLL_PRESERVE	**Preserve Register for PLL**
3:2		RG_MPPLL_TSTSEL	**Test Cicruit Input Selection**
1		RG_MPPLL_TSTCK_EN	**Test Cicruit Clock Enable** 1'b0: Monitor VDC 1'b1: Monitor Clock
0		RG_MPPLL_TST_EN	**Test Circuit Enable** 1'b0: Disable 1'b1: Enable

10010068 **DSI_PLL_PWR** **DSI PLL Power Control Register** **00000002**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								AD_DSIo_MPPLL_SDM_PWR_ACK							DA_DSIo_MPPLL_SDM_ISO_EN	DA_DSIo_MPPLL_SDM_PWR_ON
Type								RO							RW	RW
Reset								0							1	0

Bit(s)	Mnemonic	Name	Description
8		AD_DSIo_MPPLL_SDM_PWR_ACK	**PLL Power-on Acknowledge** 1'bo: No effect 1'b1: Power-on ack
1		DA_DSIo_MPPLL_SDM_ISO_EN	**PLL Isolation Enable** 1'bo: Disable 1'b1: Enable
o		DA_DSIo_MPPLL_SDM_PWR_ON	**PLL Power-on Control** 1'bo: power-off 1'b1: power-on

10010070 DSI_RGS DSI RGS Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	o
Name																**RGS_DSI_LNT_IMP_CAL_OUTPUT**
Type																RO
Reset																o

Bit(s)	Mnemonic	Name	Description
o		RGS_DSI_LNT_IMP_CAL_OUTPUT	**Watchdog action flag**

10010074 DSI_GPIO_EN DSI GPIO Enable Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	o
Name					**RG_DSIo_GPI_DR**	**RG_DSIo_GPI_SM**	**RG_DSIo_GPI9_E**	**RG_DSIo_GPI8_E**	**RG_DSIo_GPI7_E**	**RG_DSIo_GPI6_E**	**RG_DSIo_GPI5_E**	**RG_DSIo_GPI4_E**	**RG_DSIo_GPI3_E**	**RG_DSIo_GPI2_E**	**RG_DSIo_GPI1_E**	**RG_DSIo_GPIo_E**

10010074 **DSI_GPIO_EN** **DSI GPIO Enable Register** **00000000**

					IVE_EN	T_EN	N	N	N	N	N	N	N	N	N	N
Type					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset					0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
11		RG_DSI0_GPI_DRIVE_EN	**DSI GPI Driving Strength enable** 1'b0:Normal 1'b1: Enlarge
10		RG_DSI0_GPI_SMT_EN	**DSI GPI Schmit enable** 1'b0:Disable 1'b1: Enable
9		RG_DSI0_GPI9_EN	**DSI GPI9 enable, mux with PAD_TDN3** 1'b0:Disable 1'b1: Enable
8		RG_DSI0_GPI8_EN	**DSI GPI8 enable, mux with PAD_TDP3** 1'b0:Disable 1'b1: Enable
7		RG_DSI0_GPI7_EN	**DSI GPI7 enable, mux with PAD_TDN2** 1'b0:Disable 1'b1: Enable
6		RG_DSI0_GPI6_EN	**DSI GPI6 enable, mux with PAD_TDP2** 1'b0:Disable 1'b1: Enable
5		RG_DSI0_GPI5_EN	**DSI GPI5 enable, mux with PAD_TCN** 1'b0:Disable 1'b1: Enable
4		RG_DSI0_GPI4_EN	**DSI GPI4 enable, mux with PAD_TCP** 1'b0:Disable 1'b1: Enable
3		RG_DSI0_GPI3_EN	**DSI GPI3 enable, mux with PAD_TDN1** 1'b0:Disable 1'b1: Enable
2		RG_DSI0_GPI2_EN	**DSI GPI2 enable, mux with PAD_TDP1** 1'b0:Disable 1'b1: Enable
1		RG_DSI0_GPI1_EN	**DSI GPI1 enable, mux with PAD_TDN0** 1'b0:Disable 1'b1: Enable
0		RG_DSI0_GPI0_EN	**DSI GPI0 enable, mux with PAD_TDP0** 1'b0:Disable 1'b1: Enable

1001007C **DSI_PHY_SEL** **DSI MIPI TX PHY Lane Swap Selection** **00043210**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Nam										MIPI_TX_LPRX				MIPI_TX_PHYC		

1001007C **DSI_PHY_SEL** **DSI MIPI TX PHY Lane Swap Selection** **00043210**

Name										_SEL				_SEL		
Type										RW				RW		
Reset										0	0	0		1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name		MIPI_TX_PHY3_SEL				MIPI_TX_PHY2_SEL				MIPI_TX_PHY1_SEL				MIPI_TX_PHY0_SEL		
Type		RW				RW				RW				RW		
Reset		0	1	1		0	1	0		0	0	1		0	0	0

Bit(s)	Mnemonic	Name	Description
22:20		MIPI_TX_LPRX_SEL	**MIPI TX PHY Lane RX Selection**
18:16		MIPI_TX_PHYC_SEL	**MIPI TX PHY Lane CK Selection**
14:12		MIPI_TX_PHY3_SEL	**MIPI TX PHY Lane 3 Selection**
10:8		MIPI_TX_PHY2_SEL	**MIPI TX PHY Lane 2 Selection**
6:4		MIPI_TX_PHY1_SEL	**MIPI TX PHY Lane 1 Selection**
2:0		MIPI_TX_PHY0_SEL	**MIPI TX PHY Lane 0 Selection** 3'd0: DSI lane 0 3'd1: DSI lane 1 3'd2: DSI lane 2 3'd3: DSI lane 3 3'd4: DSI lane CK

10010080 **DSI_SW_CTRL_EN** **DSI Software Control Enable Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																SW_CTRL_EN
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0		SW_CTRL_EN	**DSI Software Control Enable** 1'b0: Disable 1'b1: Enable

10010084 DSI_SW_CTRL_CON0 DSI Software Control Configuration Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SW_LNTC_HSTX_DATA								SW_LNTC_LPRX_EN	SW_LNTC_HSTX_RDY	SW_LNTC_HSTX_OE	SW_LNTC_HSTX_PRE_OE	SW_LNTC_LPTX_N	SW_LNTC_LPTX_P	SW_LNTC_LPTX_OE	SW_LNTC_LPTX_PRE_OE
Type	RW								RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8		SW_LNTC_HSTX_DATA	**DSI CK Lane High-speed TX Data**
7		SW_LNTC_LPRX_EN	**DSI CK Lane Low-power RX Enable**
6		SW_LNTC_HSTX_RDY	**DSI CK Lane High-speed TX RDY**
5		SW_LNTC_HSTX_OE	**DSI CK Lane High-speed TX Output Enable**
4		SW_LNTC_HSTX_PRE_OE	**DSI CK Lane High-speed TX Pre Output Enable**
3		SW_LNTC_LPTX_N	**DSI CK Lane Low-power TX N channel**
2		SW_LNTC_LPTX_P	**DSI CK Lane Low-power TX P channel**
1		SW_LNTC_LPTX_OE	**DSI CK Lane Low-power TX Output Enable**
0		SW_LNTC_LPTX_PRE_OE	**DSI CK Lane Low-power TX Pre Output Enable**

10010088 DSI_SW_CTRL_CON1 DSI Software Control Configuration Register 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SW_LNT3_LPRX_EN	SW_LNT3_HSTX_RDY	SW_LNT3_HSTX_OE	SW_LNT3_HSTX_PRE_OE	SW_LNT3_LPTX_DN	SW_LNT3_LPTX_DP	SW_LNT3_LPTX_OE	SW_LNT3_LPTX_PRE_OE	SW_LNT2_LPRX_EN	SW_LNT2_HSTX_RDY	SW_LNT2_HSTX_OE	SW_LNT2_HSTX_PRE_OE	SW_LNT2_LPTX_DN	SW_LNT2_LPTX_DP	SW_LNT2_LPTX_OE	SW_LNT2_LPTX_PRE_OE
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

10010088 **DSI_SW_CTRL_CON1** **DSI Software Control Configuration Register 1** **00000000**

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SW_LNT1_LPRX_EN	SW_LNT1_HSTX_RDY	SW_LNT1_HSTX_OE	SW_LNT1_HSTX_PRE_OE	SW_LNT1_LPTX_N	SW_LNT1_LPTX_P	SW_LNT1_LPTX_OE	SW_LNT1_LPTX_PRE_OE	SW_LNT0_LPRX_EN	SW_LNT0_HSTX_RDY	SW_LNT0_HSTX_OE	SW_LNT0_HSTX_PRE_OE	SW_LNT0_LPTX_N	SW_LNT0_LPTX_P	SW_LNT0_LPTX_OE	SW_LNT0_LPTX_PRE_OE
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31		SW_LNT3_LPRX_EN	**DSI Data Lane 3 LPRX Enable**
30		SW_LNT3_HSTX_RDY	**DSI Data Lane 3 HSTX Ready**
29		SW_LNT3_HSTX_OE	**DSI Data Lane 3 HSTX OE**
28		SW_LNT3_HSTX_PRE_OE	**DSI Data Lane 3 HSTX Pre OE**
27		SW_LNT3_LPTX_DN	**DSI Data Lane 3 LPTX DN**
26		SW_LNT3_LPTX_DP	**DSI Data Lane 3 LPTX DP**
25		SW_LNT3_LPTX_OE	**DSI Data Lane 3 LPTX OE**
24		SW_LNT3_LPTX_PRE_OE	**DSI Data Lane 3 LPTX Pre OE**
23		SW_LNT2_LPRX_EN	**DSI Data Lane 2 LPRX Enable**
22		SW_LNT2_HSTX_RDY	**DSI Data Lane 2 HSTX Ready**
21		SW_LNT2_HSTX_OE	**DSI Data Lane 2 HSTX OE**
20		SW_LNT2_HSTX_PRE_OE	**DSI Data Lane 2 HSTX Pre OE**
19		SW_LNT2_LPTX_DN	**DSI Data Lane 2 LPTX DN**
18		SW_LNT2_LPTX_DP	**DSI Data Lane 2 LPTX DP**
17		SW_LNT2_LPTX_OE	**DSI Data Lane 2 LPTX OE**
16		SW_LNT2_LPTX_PRE_OE	**DSI Data Lane 2 LPTX Pre OE**
15		SW_LNT1_LPRX_EN	**DSI Data Lane 1 LPRX Enable**
14		SW_LNT1_HSTX_RDY	**DSI Data Lane 1 High-speed RDY**

Bit(s)	Mnemonic	Name	Description
13		SW_LNT1_HSTX_OE	DSI Data Lane 1 High-speed TX Output Enable
12		SW_LNT1_HSTX_PRE_OE	DSI Data Lane 1 High-speed TX Pre Output Enable
11		SW_LNT1_LPTX_N	DSI Data Lane 1 Low-power TX N channel
10		SW_LNT1_LPTX_P	DSI Data Lane 1 Low-power TX P channel
9		SW_LNT1_LPTX_OE	DSI Data Lane 1 Low-power TX Output Enable
8		SW_LNT1_LPTX_PRE_OE	DSI Data Lane 1 Low-power TX Pre Output Enable
7		SW_LNT0_LPRX_EN	DSI Data Lane 0 Low-power RX Enable
6		SW_LNT0_HSTX_RDY	DSI Data Lane 0 High-speed RDY
5		SW_LNT0_HSTX_OE	DSI Data Lane 0 High-speed TX Output Enable
4		SW_LNT0_HSTX_PRE_OE	DSI Data Lane 0 High-speed TX Pre Output Enable
3		SW_LNT0_LPTX_N	DSI Data Lane 0 Low-power TX N channel
2		SW_LNT0_LPTX_P	DSI Data Lane 0 Low-power TX P channel
1		SW_LNT0_LPTX_OE	DSI Data Lane 0 Low-power TX Output Enable
0		SW_LNT0_LPTX_PRE_OE	DSI Data Lane 0 Low-power TX Pre Output Enable

1001008C **DSI_SW_CTRL_CON2** **DSI Software Control Configuration Register 2** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									SW_LNTD_HSTX_DATA							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
7:0		SW_LNTD_HSTX_DATA	DSI Data Lane High-speed TX Data

10010090 **DSI_DBG_CON** **DSI Debug Control Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														MIPI_TX_DBG_OUT_EN	MIPI_TX_DBG_SEL	
Type														RW	RW	
Reset														0	0	0

Bit(s)	Mnemonic	Name	Description
2		MIPI_TX_DBG_OUT_EN	**Enable debug output of MIPI probe bus**
1:0		MIPI_TX_DBG_SEL	**Selection of debug mux of MIPI probe bus**

18 HIFSYS Control

18.1 Introduction

This module collects sub-system's all global configuration registers. The hard-wire output is used to control the hardware (EX. CG enable or clock MUX selection…etc.). And the hard-wire input is used to collect the hardware status output (EX. ID or monitor output…etc.).

18.2 Feature List

- Sub-system global configuration
- Sub-system clock MUX and CG control
- Sub-system SW reset control
- Sub-system MEMO

18.3 Block Diagram

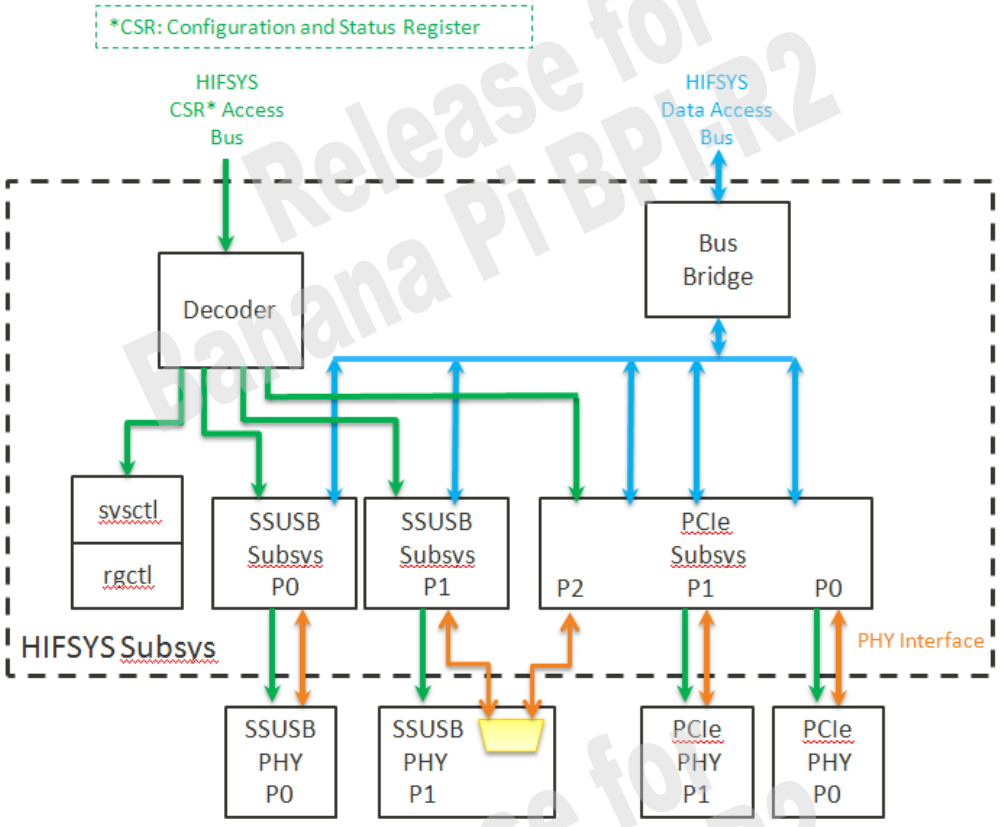


Figure 18-1.HIFSYS Block Diagram

18.4 Register Definition

HIFSYS_SYSCTLModule name: HIFSYS_SYSCTL Base address: (+1A0000000h)

Address	Name	Width	Register Function
1A000000	ID0_3	32	ID ASCII Character 0-3
1A000004	ID4_7	32	ID ASCII Character 4-7
1A000014	SYSCFG1	32	System Configuration Register 1
1A00002C	CLKCFG0	32	Clock Configuration Register 0
1A000030	CLKCFG1	32	Clock Configuration Register 1
1A000034	RSTCTL	32	Reset Control Register
1A000038	RSTSTAT	32	Reset Status Register

1A000000 ID0_3 ID ASCII Character 0-3 3637544D

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ID3								ID2							
Type	RO								RO							
Reset	0	0	1	1	0	1	1	0	0	0	1	1	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ID1								ID0							
Type	RO								RO							
Reset	0	1	0	1	0	1	0	0	0	1	0	0	1	1	0	1

Bit(s)	Name	Description
31:24	ID3	ASCII Identification Character 3
23:16	ID2	ASCII Identification Character 2
15:8	ID1	ASCII Identification Character 1
7:0	ID0	ASCII Identification Character 0

1A000004 ID4_7 ID ASCII Character 4-7 20203332

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ID7								ID6							
Type	RO								RO							
Reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ID5								ID4							
Type	RO								RO							
Reset	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0

Bit(s)	Name	Description
31:24	ID7	ASCII Identification Character 7
23:16	ID6	ASCII Identification Character 6
15:8	ID5	ASCII Identification Character 5

Bit(s)	Name	Description
7:0	ID4	**ASCII Identification Character 4**

1A000014 SYSCFG1 System Configuration Register 1 00110114

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV					CSR_PCIE_DBG_IDX_MSB2	CSR_PCIE_DBG_IDX_MSB1	CSR_PCIE_DBG_IDX_MSB0	p1_ssusb_i2c_mode	p1_ssusb_pcie_mode_sel	p1_ssusb_phy_mode		po_ssusb_i2c_mode	po_ssusb_pcie_mode_sel	po_ssusb_phy_mode	
Type	RW					RW	RW	RW	RW	RW	RW		RW	RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								PCIE_RC_MODE	NSTICKY_BY_SRST_DIS		AXI_MST_SYNC_SEL		AXI_SLV_SYNC_SEL			AXI_SYNC_MODE
Type								RW	RW		RW		RW			RW
Reset								1	0		0	1	0	1		0

Bit(s)	Name	Description
31:27	RESV	**Reserved**
26	CSR_PCIE_DBG_IDX_MSB2	**MSB of PCIe Debug Index for port-2** 0: MSB is 0 1: MSB is 1
25	CSR_PCIE_DBG_IDX_MSB1	**MSB of PCIe Debug Index for port-1** 0: MSB is 0 1: MSB is 1
24	CSR_PCIE_DBG_IDX_MSB0	**MSB of PCIe Debug Index for port-0** 0: MSB is 0 1: MSB is 1
23	p1_ssusb_i2c_mode	**PHY register access interface control. It should be set to 1 on FT pattern execution.** 1: The PHY register is accessed by I2C. 0: The PHY register is accessed by AHB/RISC
22	p1_ssusb_pcie_mode_sel	**Decide PCIe is host or device** 0: PCIe Host 1: PCIe Device
21:20	p1_ssusb_phy_mode	**Decide the PHYD is assigned to PCIe or USB3 through chip level trapping or register** 00: PCIe 01: USB3
19	p0_ssusb_i2c_mode	**PHY register access interface control. It should be set to 1 on FT pattern execution.** 1: The PHY register is accessed by I2C. 0: The PHY register is accessed by AHB/RISC

Bit(s)	Name	Description
18	po_ssusb_pcie_mode_sel	**Decide PCIe is host or device** 0: PCIe Host 1: PCIe Device
17:16	po_ssusb_phy_mode	**Decide the PHYD is assigned to PCIe or USB3 through chip level trapping or register** 00: PCIe 01: USB3
8	PCIE_RC_MODE	**PCIe Mode** 1: Root Complex mode 0: End Point mode
7	NSTICKY_BY_SRST_DIS	**Determine if Non-sticky Reset controlled by PCIe Soft Reset** 1: Non-sticky Reset NOT controlled by PCIe Soft Reset 0: Non-sticky Reset controlled by PCIe Soft Reset
5:4	AXI_MST_SYNC_SEL	**AXI aslice master direction AFIFO synchronizer selection** 0: 1 DFF SYNC 1: 2 DFF SYNC 2: 3 DFF SYNC 3: 3 DFF SYNC
3:2	AXI_SLV_SYNC_SEL	**AXI aslice slave direction AFIFO synchronizer selection** 0: 1 DFF SYNC 1: 2 DFF SYNC 2: 3 DFF SYNC 3: 3 DFF SYNC
0	AXI_SYNC_MODE	**AXI ASYNC slice configuration** 0: AXI bus in ASYNC 1: AXI bus in SYNC

1A00002C **CLKCFG0** **Clock Configuration Register 0** **C4000F00**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	OSC_1US_DIV															
Type	RW															
Reset	1	1	0	0	0	1										
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					CSR_TRGT_HI_PERF_EN	PCIE2_CLK_SEL	PCIE1_CLK_SEL	PCIE0_CLK_SEL								
Type					RW	RW	RW	RW								
Reset					1	1	1	1								

Bit(s)	Name	Description
31:26	OSC_1US_DIV	**Oscillator 1 usec Divider** Sets the maximum for the reference clock counter for either a 20 MHz or 40 MHz external XTAL input. The count increments each 1usec (indicating 1 MHz), up to the maximum, before resetting to zero. This counts the frequency of an external XTAL. This count is used to output a 32 KHz frequency to the REFCLKO pin. 0: Automatically generates a 1 usec system tick regardless of whether XTAL

Bit(s)	Name	Description
		frequency is 20 MHz or 40 MHz. 49: Default value for a reference 50 MHz clock. 39: Default value for an external 40 MHz XTAL. 19: Default value for an external 20 MHz XTAL. Others: Manual mode for tick generation.
11	CSR_TRGT_HI_PERF_EN	**xp_trgt2rb High Performance Mode Enable** 0: disable 1: enable
10	PCIE2_CLK_SEL	**PCIe2 clock selection** 1: from PCIe PHY 0: from ETHPLL 125MHz
9	PCIE1_CLK_SEL	**PCIe1 clock selection** 1: from PCIe PHY 0: from ETHPLL 125MHz
8	PCIE0_CLK_SEL	**PCIe0 clock selection** 1: from PCIe PHY 0: from ETHPLL 125MHz

1A000030 **CLKCFG1** **Clock Configuration Register 1** **07600000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						PCIE2_CLK_EN	PCIE1_CLK_EN	PCIE0_CLK_EN		USB1_HOST_PHY_CLK_EN	USB0_HOST_PHY_CLK_EN					
Type						RW	RW	RW		RW	RW					
Reset						1	1	1		1	1					
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
26	PCIE2_CLK_EN	**PCIE2 clock control** 1: Clock Enable 0: Clock Disable
25	PCIE1_CLK_EN	**PCIE1 clock control** 1: Clock Enable 0: Clock Disable
24	PCIE0_CLK_EN	**PCIE0 clock control** 1: Clock Enable 0: Clock Disable
22	USB1_HOST_PHY_CLK_EN	**USB0 host and PHY clock control** 1: Clock Enable 0: Clock Disable
21	USB0_HOST_PHY_	**USB1 host and PHY clock control**

Bit(s)	Name	Description
	CLK_EN	1: Clock Enable 0: Clock Disable

1A000034 **RSTCTL** **Reset Control Register** **07000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						PCIE2_RST	PCIE1_RST	PCIE0_RST		UPHY1_RST	UPHY0_RST					
Type						RW	RW	RW		RW	RW					
Reset						1	1	1		0	0					
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name												UHOST1_RST	UHOST0_RST			SYS_RST
Type												RW	RW			W1C
Reset												0	0			0

Bit(s)	Name	Description
26	PCIE2_RST	**PCIE2 reset control** 1: Reset Assert 0: Reset Deassert
25	PCIE1_RST	**PCIE1 reset control** 1: Reset Assert 0: Reset Deassert
24	PCIE0_RST	**PCIE0 reset control** 1: Reset Assert 0: Reset Deassert
22	UPHY1_RST	**USB PHY1 reset control** 1: Reset Assert 0: Reset Deassert
21	UPHY0_RST	**USB PHY0 reset control** 1: Reset Assert 0: Reset Deassert
4	UHOST1_RST	**USB host1 reset control** 1: Reset Assert 0: Reset Deassert
3	UHOST0_RST	**USB host0 reset control** 1: Reset Assert 0: Reset Deassert
0	SYS_RST	**Whole Sub-System Reset Control** 1: Whole System Reset 0: NA

1A000038 **RSTSTAT** **Reset Status Register** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										PCIE0_LD_RST	PCIE1_LD_RST	PCIE2_LD_RST		SWSYSRST		
Type										W1C	W1C	W1C		W1C		
Reset										0	0	0		0		

Bit(s)	Name	Description
6	PCIE0_LD_RST	**PCIe port 0 lint down reset pending register** 1: PCIe port 0 lint down reset pending 0: PCIe port 0 without lint down reset
5	PCIE1_LD_RST	**PCIe port 1 lint down reset pending register** 1: PCIe port 1 lint down reset pending 0: PCIe port 1 without lint down reset
4	PCIE2_LD_RST	**PCIe port 2 lint down reset pending register** 1: PCIe port 2 lint down reset pending 0: PCIe port 2 without lint down reset
2	SWSYSRST	**Software system reset occurred** This bit will be set if software reset the chip by writing to the RSTSYS bit in RSTCTL. Writing a '1' will clear this bit. Writing a '0' has not effect. NOTE: This register is reset only by a power on reset. 0: Has no effect. 1: Clears this bit.

19 PCie2.0

19.1 Introduction

PCIe subsys supports single Root complex (RC) with 3 Root Ports. Each Root Ports supports a Gen2 1-lane Link. PCIe subsys includes one Host/PCI bridge and 3 PCIe MAC. There are 3 bus master for data access and 1 bus slave for Configuration and Status Register (CSR) access.

19.2 Feature list

PCIe Feature	MT7623 Supported
Base Spec	Compliant to 3.0
Type	RC (3-port)
Transmission rate	Gen1/Gen2
Lane number	x1 (per port)
ASPM L0s	Not supported
PCI-PM L1/ASPM L1	Supported
L1 Substate	Not supported
L2	Not supported
OBFF	Not supported
LTR	Not supported

Table 19-1.PCIe Feature List

19.3 Block Diagram

19.3.1 PCIe Subsys (MAC)

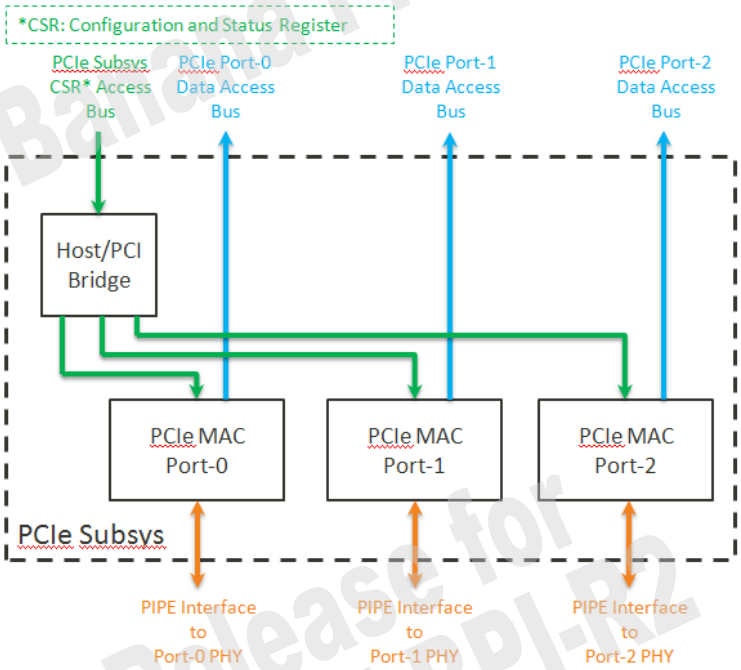


Figure 19-1.PCIe Subsys Block Diagram

19.4 Register Definition

19.4.1 pcie_express_wrap

Module name: PCI_Express Base address: (+1a140000h)

Address	Name	Width	Register Function
1a140000	PCICFG	32	PCI Configuration and Status Register
1a140008	PCIINT	32	PCI Interrupt after enable mask
1a14000c	PCIENA	32	PCI interrupt Enable
1a140020	CFGADDR	32	CONFIG TLP ADDR register
1a140024	CFGDATA	32	CONFIG TLP DATA register
1a140028	MEMBASE	32	Base Address for Memory Space Window
1a14002c	IOBASE	32	Base Address for IO Space window
1a142010	P0_BAR0SETUP	32	BAR0 Setup of PCIe0 Controller
1a142014	P0_BAR1SETUP	32	BAR1 Setup of PCIe0 Controller
1a142018	P0_IMBASEBAR0	32	Internal Memory Base Address for BAR0 Space of PCIe0
1a142030	P0_PCIE_ID	32	Vendor and Device ID of PCIe0 Controller
1a142034	P0_PCIE_CLASS	32	Class Code and Revision ID for PCIe0 Controller

Address	Name	Width	Register Function
1a142038	P0_PCIE_SUBID	32	Sub Vendor and Device ID of PCIe0 Controller(This register is valid when PCIE_RC_MODE = 0)
1a142050	P0_PCIE_SISTAT	32	PCIe0 System Info Status
1a142060	P0_DLLECR	32	PCIe0 Data Link Layer Error Counter Register
1a142064	P0_ECRCCR	32	PCIe0 ECRC Counter register
1a142070	P0_LTSSM_DELAY	32	PCIe0 LTSSM Delay
1a143010	P1_BAR0SETUP	32	BAR0 Setup of PCIe1 Controller
1a143014	P1_BAR1SETUP	32	BAR1 Setup of PCIe1 Controller
1a143018	P1_IMBASEBAR0	32	Internal Memory Base Address for BAR0 Space of PCIe1
1a143030	P1_PCIE_ID	32	Vendor and Device ID of PCIe1 Controller
1a143034	P1_PCIE_CLASS	32	Class Code and Revision ID for PCIe1 Controller
1a143038	P1_PCIE_SUBID	32	Sub Vendor and Device ID of PCIe1 Controller(This register is valid when PCIE_RC_MODE = 0)
1a143050	P1_PCIE_SISTAT	32	PCIe1 System Info Status
1a143060	P1_DLLECR	32	PCIe1 Data Link Layer Error Counter Register
1a143064	P1_ECRCCR	32	PCIe1 ECRC Counter register
1a143070	P1_LTSSM_DELAY	32	PCIe1 LTSSM Delay
1a144010	P2_BAR0SETUP	32	BAR0 Setup of PCIe2 Controller
1a144014	P2_BAR1SETUP	32	BAR1 Setup of PCIe2 Controller
1a144018	P2_IMBASEBAR0	32	Internal Memory Base Address for BAR0 Space of PCIe2
1a144030	P2_PCIE_ID	32	Vendor and Device ID of PCIe2 Controller
1a144034	P2_PCIE_CLASS	32	Class Code and Revision ID for PCIe2 Controller
1a144038	P2_PCIE_SUBID	32	Sub Vendor and Device ID of PCIe2 Controller(This register is valid when PCIE_RC_MODE = 0)
1a144050	P2_PCIE_SISTAT	32	PCIe2 System Info Status
1a144060	P2_DLLECR	32	PCIe2 Data Link Layer Error Counter Register
1a144064	P2_ECRCCR	32	PCIe2 ECRC Counter register
1a144070	P2_LTSSM_DELAY	32	PCIe2 LTSSM Delay
1a148000	PHY_RST	32	PHY Reset (P1/P0)
1a150000~1a15fffc	MEM_WIN[n] (n=0~16383)	32	PCI Memory Space Access Window
1a160000~1a16fffc	IO_WIN[n] (n=0~16383)	32	PCI IO Space Access Window
60000000~6ffffffc	MEM_DIRECT1[n] (n=0~67108863)	32	PCI Direct Access Memory Window (Host Mode)
1a800000~1abffffc	MEM_DIRECT2[n] (n=0~1048575)	32	PCI Direct Access Memory Window (iNIC Mode)

1A140000 PCICFG PCI Configuration and Status Register 021017FE

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					P2P_BR_DEVNUM2				P2P_BR_DEVNUM1				P2P_BR_DEVNUM0			

Type					RW				RW				RW			
Reset					0	0	1	0	0	0	0	1	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			CSR_ENABLE_CLKREQ	CSR_7628ECO2_MALFORMED_CPLD_EN	MAC_TXDETECTRX_FIX_EN	MULTI_DMA_EN	BURST_SIZE_P2		BURST_SIZE_P1		BURST_SIZE_P0		PCIRST2	PCIRST1	PCIRST0	
Type			RW	RW	RW	RW	RW		RW		RW		RW	RW	RW	
Reset			0	1	0	1	1	1	1	1	1	1	1	1	1	

Bit(s)	Name	Description
27:24	P2P_BR_DEVNUM2	**Device number setting of Virtual PCI-PCI bridge #2.**
23:20	P2P_BR_DEVNUM1	**Device number setting of Virtual PCI-PCI bridge #1.**
19:16	P2P_BR_DEVNUM0	**Device number setting of Virtual PCI-PCI bridge #0.**
13	CSR_ENABLE_CLKREQ	**If enabled, CLKREQ input from PAD is used**
12	CSR_7628ECO2_MALFORMED_CPLD_EN	**7628 ECO for malformed CPLD issue**
11	MAC_TXDETECTRX_FIX_EN	
10	MULTI_DMA_EN	**Multi-issue DMA Enable** 0: disable 1: enable
9:8	BURST_SIZE_P2	**Multi-issue burst size #2**
7:6	BURST_SIZE_P1	**Multi-issue burst size #1**

Bit(s)	Name	Description
5:4	BURST_SIZE_P0	**Multi-issue burst size #0**
3	PCIRST2	**PCI reset control of Port 2** Available when PCIe Controller in Host mode 1: Assert the PERST_N Pin 0: De-assert the PERST_N Pin
2	PCIRST1	**PCI reset control of Port 1** Available when PCIe Controller in Host mode 1: Assert the PERST_N Pin 0: De-assert the PERST_N Pin
1	PCIRST0	**PCI reset control of Port 0** Available when PCIe Controller in Host mode 1: Assert the PERST_N Pin 0: De-assert the PERST_N Pin

1A140008 PCIINT PCI Interrupt after enable mask 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										PCIINT2	PCIINT1	PCIINT0				
Type										RO	RO	RO				
Reset										0	0	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
22	PCIINT2	**PCIe2 interrupt input in RC mode** This bit indicates the PCIe interrupt from PCIe2 slot 1: PCIe2 slot have interrupt occur 0: PCIe2 slot have no interrupt occur
21	PCIINT1	**PCIe1 interrupt input in RC mode** This bit indicates the PCIe interrupt from PCIe1 slot 1: PCIe1 slot have interrupt occur

Bit(s)	Name	Description
		0: PCIe1 slot have no interrupt occur
20	PCIINT0	**PCIe0 interrupt input in RC mode** This bit indicates the PCIe interrupt from PCIe0 slot 1: PCIe0 slot have interrupt occur 0: PCIe0 slot have no interrupt occur

1A14000C PCIENA PCI interrupt Enable 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name										PCIINT2_EN	PCIINT1_EN	PCIINT0_EN				
Type										RW	RW	RW				
Reset										0	0	0				
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
22	PCIINT2_EN	**PCIINT (PCIe2) Enable Control** 1: Enable PCIINT (PCIe2) interrupt 0: Disable PCIINT (PCIe2) interrupt
21	PCIINT1_EN	**PCIINT (PCIe1) Enable Control** 1: Enable PCIINT (PCIe1) interrupt 0: Disable PCIINT (PCIe1) interrupt
20	PCIINT0_EN	**PCIINT (PCIe0) Enable Control** 1: Enable PCIINT (PCIe0) interrupt 0: Disable PCIINT (PCIe0) interrupt

1A140020 CFGADDR CONFIG TLP ADDR register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name					EXTREGNUM				BUSNUM							
Type					RW				RW							
Reset					0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	DEVICENUM					FUNNUM			REGNUM							
Type	RW					RW			RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Bit(s)	Name	Description
27:24	EXTREGNUM	**Extent Register Number, only avail for PCIe**
23:16	BUSNUM	**Bus Number**
15:11	DEVICENUM	**Device Number**
10:8	FUNNUM	**Function Number**
7:2	REGNUM	**Register Number**

1A140024 CFGDATA CONFIG TLP DATA register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CFG_DATA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CFG_DATA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	CFG_DATA	**CONFIG_DATA Register** Write to or read from this register will generates a Configuration Cycle in Host mode.

1A140028 MEMBASE Base Address for Memory Space Window 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMB ASE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Na																

me																
Type																
Reset																

Bit(s)	Name	Description
31:16	MEMBASE	Base Address for Memory Space Window

1A14002C IOBASE Base Address for IO Space window 1A160000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IOBA SE															
Type	RW															
Reset	0	0	0	1	1	0	1	0	0	0	0	1	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																
Type																
Reset																

Bit(s)	Name	Description
31:16	IOBASE	Base Address for IO Space Window

1A142010 P0_BAR0SETUP BAR0 Setup of PCIe0 Controller 01FF0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR0 MSK															
Type	RW															
Reset	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BAR0 ENABLE
Type																RW
Res																1

et																

Bit(s)	Name	Description
31:16	BAR0MSK	**Setup for Base Address Register BAR0** When the mask bit is '1', the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is '0', the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is '0'. *Please set this value before the CfgWr to BAR0, else the CFGWr to BAR0 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR0ENABLE	**to determind if the BAR0 space will be enabled according to BAR1MSK** 1: Enable. 0: Disable

1A142014 P0_BAR1SETUP BAR1 Setup of PCIe0 Controller 00FF8001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR1MSK															
Type	RW															
Res	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

et																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BAR1MSK															BAR1ENABLE
Type	RW															RW
Reset	1															1

Bit(s)	Name	Description
31:15	BAR1MSK	**Setup for Base Address Register BAR1** When the mask bit is 1, the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is 0, the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is 0. *Please set this value before the CfgWr to BAR1, else the CFGWr to BAR1 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR1ENABLE	**to determind if the BAR1 space will be enabled according to BAR1MSK** 1: Enable. 0: Disable

1A142018 P0_IMBASEBAR0 Internal Memory Base Address for BAR0 Space of PCIe0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IMBASEBAR 0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IMBASEBAR 0															
Type	RW															
Reset	0															

Bit(s)	Name	Description
31:15	IMBASEBAR0	**Internal Memory Base address for BAR0** This register is used when CHIP behaves as a PCI Express RC. The actually internal memory address being accessed by an external PCI host can be obtained from the following formula: CHIP address begin accessed = (PCI Address - BAR0) + IMBASEBAR0. When write to this register, the related bit will take effect when the corresponding bit in BAR0MSK bit is 1 and BAR0ENABLE is 1. Internal Memory Base address for BAR0 This register is used when CHIP behaves as a PCIe RC.

1A142030 P0_PCIE_ID Vendor and Device ID of PCIe0 Controller 080114C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVID															
Type	RW															
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VENID															
Type	RW															
Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	DEVID	**Device ID**

Bit(s)	Name	Description
15:0	VENID	**Vendor ID**

1A142034 Po_PCIE_CLASS Class Code and Revision ID for PCIe0 Controller 0D800001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CCODE															
Type	RW															
Reset	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCODE								REVID							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:8	CCODE	**Class Code**
7:0	REVID	**Revision ID**

1A142038 Po_PCIE_SUBID Sub Vendor and Device ID of PCIe0 Controller(This register is valid when PCIE_RC_MODE = 0) 762314C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SUBSYSID															
Type	RW															
Reset	0	1	1	1	0	1	1	0	0	0	1	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SUBVENID															
Type	RW															
Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	SUBSYSID	**Sub Device ID**
15:0	SUBVENID	**Sub Vendor ID**

1A142050 P0_PCIE_SISTAT PCIe0 System Info Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																LINKUP
Type																RO
Reset																0

Bit(s)	Name	Description
0	LINKUP	**PCIe Linkup Status** 1: Linkup 0: Linkdown

1A142060 P0_DLLECR PCIe0 Data Link Layer Error Counter Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DLLP_ERR_CNT	**Datalink Layer Error counter register record how many times datalink layer error happened**

1A142064 P0_ECRCCR PCIe0 ECRC Counter register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ECRC_ERR_CNT	**ECRC Error counter register record how many times ECRC error happened**

1A142070 P0_LTSSM_DELAY PCIe0 LTSSM Delay 00000F0C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					LoS_IDLE_DELAY				NFTS_TIMEOUT_DELAY							
Type					RW				RW							
Reset					1	1	1	1	0	0	0	0	1	1	0	0

Bit(s)	Name	Description
11:8	LoS_IDLE_DELAY	**Entry LoS_IDLE delay for various PHY**
7:0	NFTS_TIMEOUT_DELAY	**NFTS timeout delay for various PHY**

1A143010 P1_BAR0SETUP BAR0 Setup of PCIe1 Controller 01FF0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR0 MSK															
Type	RW															
Reset	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BAR0ENABLE
Type																RW

e																
Reset																1

Bit(s)	Name	Description
31:16	BAR0MSK	**Setup for Base Address Register BAR0** When the mask bit is '1', the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is '0', the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is '0'. *Please set this value before the CfgWr to BAR0, else the CFGWr to BAR0 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR0ENABLE	**to determind if the BAR0 space will be enabled according to BAR1MSK** 1: Enable. 0: Disable

1A143014 P1_BAR1SETUP BAR1 Setup of PCIe1 Controller 00FF8001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR1MSK															
Typ	RW															

e																
Reset	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BAR1MSK															BAR1ENABLE
Type	RW															RW
Reset	1															1

Bit(s)	Name	Description
31:15	BAR1MSK	**Setup for Base Address Register BAR1** When the mask bit is 1, the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is 0, the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is 0. *Please set this value before the CfgWr to BAR1, else the CFGWr to BAR1 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR1ENABLE	**to determind if the BAR1 space will be enabled according to BAR1MSK** 1: Enable.

Bit(s)	Name	Description
		0: Disable

1A143018 P1_IMBASEBAR0 Internal Memory Base Address for BAR0 of PCIe1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IMBASEBAR0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IMBASEBAR0															
Type	RW															
Reset	0															

Bit(s)	Name	Description
31:15	IMBASEBAR0	**Internal Memory Base address for BAR0** This register is used when CHIP behaves as a PCI Express RC. The actually internal memory address being accessed by an external PCI host can be obtained from the following formula: CHIP address begin accessed = (PCI Address - BAR0) + IMBASEBAR0. When write to this register, the related bit will take effect when the corresponding bit in BAR0MSK bit is 1 and BAR0ENABLE is 1. Internal Memory Base address for BAR0 This register is used when CHIP behaves as a PCIe RC.

1A143030 P1_PCIE_ID Vendor and Device ID of PCIe1 Controller 080114C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVID															
Type	RW															
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VENID															
Type	RW															
Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	DEVID	**Device ID**
15:0	VENID	**Vendor ID**

1A143034 P1_PCIE_CLASS Class Code and Revision ID for PCIe1 Controller 0D800001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CCODE															
Type	RW															
Reset	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCODE								REVID							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:8	CCODE	**Class Code**
7:0	REVID	**Revision ID**

1A143038 P1_PCIE_SUBID Sub Vendor and Device ID of PCIe1 Controller(This register is valid when PCIE_RC_MODE = 0) 762314C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SUBSYSID															
Type	RW															
Reset	0	1	1	1	0	1	1	0	0	0	1	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SUBVENID															
Type	RW															
Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	SUBSYSID	**Sub Device ID**
15:0	SUBVENID	**Sub Vendor ID**

1A143050 P1_PCIE_SISTAT PCIe1 System Info Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																LINKUP
Type																RO
Reset																0

Bit(s)	Name	Description
0	LINKUP	**PCIe Linkup Status** 1: Linkup 0: Linkdown

1A143060 P1_DLLECR PCIe1 Data Link Layer Error Counter Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DLLP_ERR_CNT	**Datalink Layer Error counter register record how many times datalink layer error happened**

1A143064 P1_ECRCCR PCIe1 ECRC Counter register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ECRC_ERR_CNT	**ECRC Error counter register record how many times ECRC error happened**

1A143070 P1_LTSSM_DELAY PCIe1 LTSSM Delay 00000F0C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					LoS_IDLE_DELAY				NFTS_TIMEOUT_DELAY							
Type					RW				RW							
Reset					1	1	1	1	0	0	0	0	1	1	0	0

Bit(s)	Name	Description
11:8	LoS_IDLE_DELAY	**Entry LoS_IDLE delay for various PHY**
7:0	NFTS_TIMEOUT_DELAY	**NFTS timeout delay for various PHY**

1A144010 P2_BAR0SETUP BAR0 Setup of PCIe2 Controller 01FF0001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR0 MSK															
Type	RW															
Reset	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																BAR0ENABLE

Type															RW
Reset															1

Bit(s)	Name	Description
31:16	BAR0MSK	**Setup for Base Address Register BAR0** When the mask bit is '1', the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is '0', the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is '0'. *Please set this value before the CfgWr to BAR0, else the CFGWr to BAR0 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR0ENABLE	**to determind if the BAR0 space will be enabled according to BAR1MSK** 1: Enable. 0: Disable

1A144014 P2_BAR1SETUP BAR1 Setup of PCIe2 Controller 00FF8001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	BAR1MSK															

Type	RW															
Reset	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	BAR1MSK															BAR1ENABLE
Type	RW															RW
Reset	1															1

Bit(s)	Name	Description
31:15	BAR1MSK	**Setup for Base Address Register BAR1** When the mask bit is 1, the corresponding address bit will be masked as a hit as if no address comparison has been made. When the mask bit is 0, the corresponding address bit will be used for address comparison to determine an address hit. Each base address register can be mapped from 64KB to 2GB. The mask bit will be ignored when the corresponding enable bit is 0. *Please set this value before the CfgWr to BAR1, else the CFGWr to BAR1 will get unknown result. 16'h7fff: 2G Space 16'h3fff: 1G Space 16'h1fff: 512M Space 16'h0fff: 256M Space 16'h07ff: 128M Space 16'h03ff: 64M Space 16'h01ff: 32M Space(Default) 16'h00ff: 16M Space 16'h007f: 8M Space 16'h003f: 4M Space 16'h001f: 2M Space 16'h000f: 1M Space 16'h0007: 512K Space 16'h0003: 256K Space 16'h0001: 128K Space 16'h0000: 64K Space Other: Not Support
0	BAR1ENABLE	**to determind if the BAR1 space will be enabled according to BAR1MSK**

Bit(s)	Name	Description
		1: Enable. 0: Disable

1A144018 P2_IMBASEBAR0 Internal Memory Base Address for BAR0 Space of PCIe2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IMBASEBAR0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IMBASEBAR0															
Type	RW															
Reset	0															

Bit(s)	Name	Description
31:15	IMBASEBAR0	**Internal Memory Base address for BAR0** This register is used when CHIP behaves as a PCI Express RC. The actually internal memory address being accessed by an external PCI host can be obtained from the following formula: CHIP address begin accessed = (PCI Address - BAR0) + IMBASEBAR0. When write to this register, the related bit will take effect when the corresponding bit in BAR0MSK bit is 1 and BAR0ENABLE is 1. Internal Memory Base address for BAR0 This register is used when CHIP behaves as a PCIe RC.

1A144030 P2_PCIE_ID Vendor and Device ID of PCIe2 Controller 080114C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVID															
Type	RW															
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VENID															
Type	RW															

Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	DEVID	**Device ID**
15:0	VENID	**Vendor ID**

1A144034 P2_PCIE_CLASS Class Code and Revision ID for PCIe2 Controller 0D800001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CCODE															
Type	RW															
Reset	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CCODE								REVID							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Name	Description
31:8	CCODE	**Class Code**
7:0	REVID	**Revision ID**

1A144038 P2_PCIE_SUBID Sub Vendor and Device ID of PCIe2 Controller(This register is valid when PCIE_RC_MODE = 0) 762314C3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SUBSYSID															
Type	RW															
Reset	0	1	1	1	0	1	1	0	0	0	1	0	0	0	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SUBVENID															
Type	RW															
Reset	0	0	0	1	0	1	0	0	1	1	0	0	0	0	1	1

Bit(s)	Name	Description
31:16	SUBSYSID	**Sub Device ID**
15:0	SUBVENID	**Sub Vendor ID**

Bit(s)	Name	Description

1A144050 P2_PCIE_SISTAT PCIe2 System Info Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																LINKUP
Type																RO
Reset																0

Bit(s)	Name	Description
0	LINKUP	**PCIe Linkup Status** 1: Linkup 0: Linkdown

1A144060 P2_DLLECR PCIe2 Data Link Layer Error Counter Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DLLP_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	DLLP_ERR_CNT	**Datalink Layer Error counter register record how many times datalink layer error happened**

1A144064 P2_ECRCCR PCIe2 ECRC Counter register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ECRC_ERR_CNT															
Type	RU															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	ECRC_ERR_CNT	**ECRC Error counter register record how many times ECRC error happened**

1A144070 P2_LTSSM_DELAY PCIe2 LTSSM Delay 00000F0C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					LoS_IDLE_DELAY				NFTS_TIMEOUT_DELAY							
Type					RW				RW							
Reset					1	1	1	1	0	0	0	0	1	1	0	0

Bit(s)	Name	Description
11:8	LoS_IDLE_DELAY	**Entry LoS_IDLE delay for various PHY**
7:0	NFTS_TIMEOUT_DELAY	**NFTS timeout delay for various PHY**

1A148000 PHY_RST PHY Reset (P1/P0) 00000003

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name															P1_PHY_RST	P0_PHY_RST
Type															RW	RW
Reset															1	1

Bit(s)	Name	Description
1	P1_PHY_RST	**Reset for PCIE P1 PHY** 0: Assert reset 1: De-assert reset
0	P0_PHY_RST	**Reset for PCIE P0 PHY** 0: Assert reset 1: De-assert reset

1A150000~ 1A15FFFC **MEM_WIN[n](n=0~16383)** **PCI Memory Space Access Window** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEMWIN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEMWIN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEMWIN	**Writing to this register will initiate a bus master write access to an external PCI device memory space; reading this register will initiate a bus master read access to an external PCI device memory space. The address accessed is specified as requested address 0+MEMBASE if MEMWIN0000 is accessed, 4+MEMBASE if MEMWIN0004 is accessed, etc.** Note: This register is only used when the PCI core is functioning as a master.

1A160000~ IO_WIN **PCI IO Space Access Window** **00000000**

1A16FFFC [n](n=0~16383)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IOWIN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IOWIN															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	IOWIN	Writing to this register will initiate a bus master write access to an external PCI device s IO space; reading this register will initiate a bus master read access to an external PCI device IO space. The address accessed is specified as requested address 0 IOBASE if IOWIN0000 is accessed, 4+IOBASE if IOWIN0004 is accessed, etc. Note: This register is only used when the PCI core is functioning as a master.

60000000~6FFFFFFC MEM_DIRECT1 [n](n=0~67108863) PCI Direct Access Memory Window (Host Mode) 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEM_DIRECT1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEM_DIRECT1															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEM_DIRECT1	**Writing to this register will initiate a bus master write access to an external PCI device memory space; reading this register will initiate a bus master read access to an external PCI device memory space. This is direct access window, no base address setting and address translation in this window.**

1A800000~1ABFFFFC MEM_DIRECT2 [n](n=0~1048575) PCI Direct Access Memory Window (iNIC Mode) 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MEM_DIRECT2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MEM_DIRECT2															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MEM_DIRECT2	**Writing to this register will initiate a bus master write access to an external PCI device memory space; reading this register will initiate a bus master read access to an external PCI device memory space. This is direct access window, no base address setting and address translation in this window.**

20 USB3.0

20.1 Introduction

The SSUSB IP within a chip is composed by three parts: the PHYA macro, the U2 PHYD macro and SSUSB MAC in RTL. Based on the USB specification, USB3 port should have downward compatibility to USB2 devices. Thus the definition of SSUSB IP should contain one U3 PHY for Super Speed connection and U2 PHY for High Speed, Full Speed and Low Speed connection. This chip supports 2 full set SSUSB with Port0 and Port1, the configration of these two sets is same and dedicated.
there is multiple ports configuration with USB3 Host and USB2 Host. Each PHY has its own MAC to handle protocol packets.
All the resources of endpoints are shared by all ports and handled by xHCI controller. Firmware could dynamically allocate resource for different ports and turn on/off each port separately. With the high transmission rate, up to 5-Gbs Tx and 5-Gbs Rx, the DMA channel of U3 port is separated from U2 port but it still could be multiplexed on system.

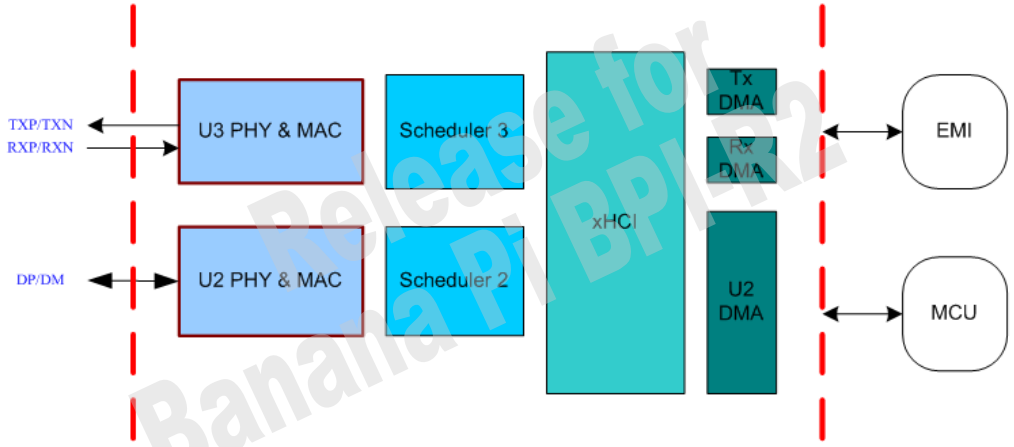


Figure 20-1.SSUSB Host Architecture

20.2 Feature List

- Lower power management (LPM) on U2 port
- U0/U1/U2/U3 state on U3 port
- Dedicated DMA channel for USB3 data transfer
- Support USB3 port with 5-Gbs Tx and 5-Gbx Rx bandwidth
- Support USB2 port with dedicated 480-Mbs bandwidth and DMA
- Control / Bulk / Interrupt / Isochronous PIPE type
- Compatible to connect to U2/U3 Hub
- Smart scheduling algorithm
- Up to 15 Devices per set
- Up to 32 Endpoints support per setBlock Diagram

20.2.1 SSUSB Subsyss (MAC)

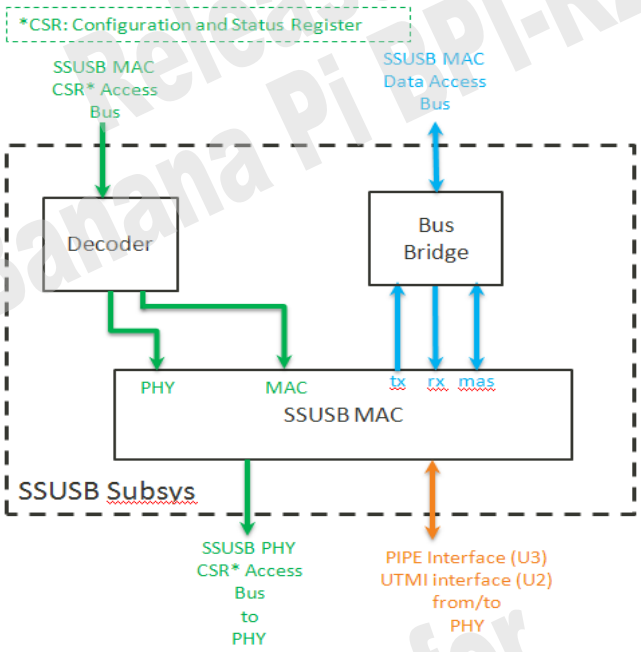


Figure 20-2.SSUSB Subsys Block Diagram

20.3 Register Definition

20.3.1 ssusb2_xhci_exclude_port_csr

Module name: ssusb2_xhci_exclude_port_csr Base address: (+1a240000h)

Address	Name	Width	Register Function
1a240000	CAPLENGTH	32	Capability Register Length
1a240004	HCSPARAMS1	32	Structural Parameters 1
1a240008	HCSPARAMS2	32	Structural Parameters 2
1a24000c	HCSPARAMS3	32	Structural Parameters 3
1a240010	HCCPARAMS	32	Capability Parameters
1a240014	DBSOFF	32	Doorbell Offset
1a240018	RTSOFF	32	Runtime Register Space Offset
1a240020	USBCMD	32	USB Command
1a240024	USBSTS	32	USB Status
1a240028	PAGESIZE	32	Page Size
1a240034	DNCTRL	32	Device Notification Control
1a240038	CRCR1	32	Command Ring Control 1
1a24003c	CRCR2	32	Command Ring Control 2
1a240050	DCBAAP_LO	32	Device Context Base Address Array Pointer Lo
1a240054	DCBAAP_HI	32	Device Context Base Address Array Pointer High
1a240058	CONFIG	32	Configure
1a240500	SUPP_PTCL_REG1	32	xHCI Supported Protocol Capability 1 Register

Address	Name	Width	Register Function
1a240504	SUPP_PTCL_RE G2	32	xHCI Supported Protocol Capability 2 Register
1a240508	SUPP_PTCL_RE G3	32	xHCI Supported Protocol Capability 3 Register
1a240510	SUPP_PTCL_RE G4	32	xHCI Supported Protocol Capability 1 Register
1a240514	SUPP_PTCL_RE G5	32	xHCI Supported Protocol Capability 2 Register
1a240518	SUPP_PTCL_RE G6	32	xHCI Supported Protocol Capability 3 Register
1a240600	MFINDEX	32	Microframe Index
1a240620	IMAN	32	Interrupter Management
1a240624	IMOD	32	Interrupter Moderation
1a240628	ERSTSZ	32	Event Ring Segment Table Size
1a240630	ERSTBA_LO	32	Event Ring Segment Table Base Address Lo
1a240634	ERSTBA_HI	32	Event Ring Segment Table Base Address Hi
1a240638	ERDP_LO	32	Event Ring Segment Table Base Address Lo
1a24063c	ERDP_HI	32	Event Ring Segment Table Base Address Hi
1a240800	HOST_CMD_DB	32	Host Controller Doorbell Registers
1a240804	DEVICE1_DB	32	Device 1 Doorbell Registers
1a240808	DEVICE2_DB	32	Device 2 Doorbell Registers
1a24080c	DEVICE3_DB	32	Device 3 Doorbell Registers
1a240810	DEVICE4_DB	32	Device 4 Doorbell Registers
1a240814	DEVICE5_DB	32	Device 5 Doorbell Registers
1a240818	DEVICE6_DB	32	Device 6 Doorbell Registers
1a24081c	DEVICE7_DB	32	Device 7 Doorbell Registers
1a240820	DEVICE8_DB	32	Device 8 Doorbell Registers
1a240824	DEVICE9_DB	32	Device 9 Doorbell Registers
1a240828	DEVICE10_DB	32	Device 10 Doorbell Registers
1a24082c	DEVICE11_DB	32	Device 11 Doorbell Registers
1a240830	DEVICE12_DB	32	Device 12 Doorbell Registers
1a240834	DEVICE13_DB	32	Device 13 Doorbell Registers
1a240838	DEVICE14_DB	32	Device 14 Doorbell Registers
1a24083c	DEVICE15_DB	32	Device 15 Doorbell Registers
1a240900	HSRAM_DBGCTL	32	Host SRAM Debug Control Register
1a240904	HSRAM_DBGMOD E	32	Host SRAM Debug Mode Register
1a240908	HSRAM_DBGSEL	32	Host SRAM Debug Select Register
1a24090c	HSRAM_DBGADR	32	Host SRAM Debug Address Register
1a240910	HSRAM_DBGDR	32	Host SRAM Debug Data Register
1a240920	HSRAM_DELSEL_0	32	Host SRAM Delay Select 0
1a240924	HSRAM_DELSEL_1	32	Host SRAM Delay Select 1
1a240930	LS_EOF	32	Low Speed EOF Start Offset
1a240934	FS_EOF	32	Full Speed EOF Start Offset
1a240938	SYNC_HS_EOF	32	Synchronous High Speed EOF Start Offset
1a24093c	SS_EOF	32	Super Speed EOF Start Offset
1a240940	SOF_OFFSET	32	SOF Offset
1a240944	HFCNTR_CFG	32	Host Frame Counter Configuration
1a240948	XACT3_CFG	32	Super Speed Transaction Configuration
1a24094c	XACT2_CFG	32	USB2 Transaction Configuration

Address	Name	Width	Register Function
1a240950	HDMA_CFG	32	Host DMA Configuration
1a240954	ASYNC_HS_EOF	32	Asynchronous High Speed EOF Start Offset
1a240958	AXI_WR_DMA_CFG	32	AXI WR DMA configuration register.
1a24095c	AXI_RD_DMA_CFG	32	AXI RD DMA configuration register.
1a240960	HSCH_CFG1	32	Host Scheduler Configuration Register 1
1a240964	CMD_CFG	32	Command Configuration
1a240968	EP_CFG	32	Endpoint Status Configuration
1a24096c	EVT_CFG	32	Event Configuration
1a240970	TRBQ_CFG	32	TRBQ Configuration
1a240974	U3PORT_CFG	32	USB3 Port Configuration
1a240978	U2PORT_CFG	32	USB2 Port Configuration
1a24097c	HSCH_CFG2	32	Host Scheduler Configuration Register 2
1a240980	SW_ERDY	32	Software ERDY
1a2409a0	SLOT_EP_STS0	32	Slot and EP Resource Status0
1a2409a4	SLOT_EP_STS1	32	Slot and EP Resource Status1
1a2409a8	SLOT_EP_STS2	32	Slot and EP Resource Status2
1a2409b0	RST_CTRL0	32	Host reset control Register 2
1a2409b4	RST_CTRL1	32	Host reset control Register 3
1a2409f0	SPARE0	32	Spare Register 0
1a2409f4	SPARE1	32	Spare Register 1

1A240000 CAPLENGTH Capability Register Length 00960020

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	HCIVERSION															
Type	RO															
Reset	0	0	0	0	0	0	0	0	1	0	0	1	0	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									CAPLENGTH							
Type									RO							
Reset									0	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		HCIVERSION	**This is a two-byte register containing a BCD encoding of the xHCI specification revision number supported by this host controller. The most significant byte of this register represents a major revision and the least significant byte is the minor revision. e.g. 0100h corresponds to xHCI version 1.0.** Note: Pre-release versions of the xHC shall declare the specific version of the xHCI that it was implemented against. e.g. 0090h = version 0.9.

Bit(s)	Mnemonic	Name	Description
7:0		CAPLENGTH	**This register is used as an offset to add to register base to find the beginning of the Operational Register Space**

1A240004 HCSPARAMS1 Structural Parameters 1 0700010F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAXPORTS													MAXINTRS		
Type	RO													RO		
Reset	0	0	0	0	0	1	1	1						0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAXINTRS								MAXSLOTS							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:24		MAXPORTS	**Number of Ports (MaxPorts).** This field specifies the number of physical downstream ports implemented on this host controller. The value of this field determines how many port registers are addressable in the Operational Register Space (refer to Table 26). Valid values are in the range of 1h to FFh.
18:8		MAXINTRS	**Number of Interrupters (MaxIntrs).** This field specifies the number of Interrupters implemented on this host controller. Each Interrupter is allocated to a vector of MSI-X and controls its generation and moderation. The value of this field determines how many Interrupter Register Sets are addressable in the Runtime Register Space (refer to section 5.5). Valid values are in the range of 1h to 400h. A '0' in this field is undefined.
7:0		MAXSLOTS	**Number of Device Slots (MaxSlots).** This field specifies the maximum number of Device Context Structures and Doorbell Array entries this host controller can support. Valid values are in the range of 1 to 255. The value of '0' is reserved.

1A240008 HCSPARAMS2 Structural Parameters 2 00001004

Bit	31	30	29	2	2	26	25	24	23	22	21	20	19	18	17	16

				8	7											
Name	MAX_SCRATCHPAD_BUFS					SPR										
Type	RO					RO										
Reset	0	0	0	0	0	0										
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				IOC_INTERVAL					ERST_MAX				IST			
Type				RO					RO				RW			
Reset				1	0	0	0	0	0	0	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
31:27		MAX_SCRATCHPAD_BUFS	**Max Scratchpad Buffers (Max Scratchpad Bufs).** Default = implementation dependent. Valid values are 0-31. This field indicates the number of Scratchpad Buffers system software shall reserve for the xHC. Refer to section 4.20 for more information.
26		SPR	**Scratchpad Restore (SPR).** Default = implementation dependent. If Max Scratchpad Buffers is > '0' then this flag indicates whether the xHC uses the Scratchpad Buffers for saving state when executing Save and Restore State operations. If Max Scratchpad Buffers is = '0' then this flag shall be '0'. Refer to section 4.23.2 for more information. A value of '1' indicates that the xHC requires the integrity of the Scratchpad Buffer space to be maintained across power events. A value of '0' indicates that the Scratchpad Buffer space may be freed and reallocated between power events.
12:8		IOC_INTERVAL	**IOC Interval.** Default = implementation dependent. Valid values are 0 - 23. This field determines the maximum frequency with which the IOC flag may be set for this xHC implementation. The IOC flag may be set in TRBs that define data buffers that are more than 2 IOC Interval bytes apart. Refer to section 4.11.7.1 for more information on the use and interpretation of the IOC Interval.
7:4		ERST_MAX	**Event Ring Segment Table Max (ERST Max).** Default = implementation dependent. Valid values are

Bit(s)	Mnemonic	Name	Description
			0 - 15. This field determines the maximum value supported the Event Ring Segment Table Base Size registers (5.5.2.3.1), where: The maximum number of Event Ring Segment Table entries = 2 ERST Max. e.g. if the ERST Max = 7, then the xHC Event Ring Segment Table(s) supports up to 128 entries, 15 then 32K entries, etc.
3:0		IST	**Isochronous Scheduling Threshold (IST).** Default = implementation dependent. The value in this field indicates to system software the minimum distance (in time) that it is required to stay ahead of the host controller while adding TRBs, in order to have the host controller process them at the correct time. The value shall be specified in terms of number of frames/microframes. If bit [3] of IST is cleared to '0', software can add a TRB no later than IST[2:0] Microframes before that TRB is scheduled to be executed. If bit [3] of IST is set to '1', software can add a TRB no later than IST[2:0] Frames before that TRB is scheduled to be executed. Refer to Section 4.14.2 for details on how software uses this information for scheduling isochronous transfers.

1A24000C HCSPARAMS3 Structural Parameters 3 00010001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	U2_DEVICE_EXIT_LATENCY															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									U1_DEVICE_EXIT_LATENCY							
Type									RW							
Reset									0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
31:16		U2_DEVICE_EXIT_LATENCY	**U2 Device Exit Latency.** Worst case latency to transition from U2 to U0. Applies to all root hub ports. The following are permissible values: Value Description 0000h Zero 0001h Less than 1 us.

Bit(s)	Mnemonic	Name	Description
			0002h Less than 2 us. ... 07FFh Less than 2047 us. 0800-FFFFh Reserved
7:0		U1_DEVICE_EXIT_LATENCY	**U1 Device Exit Latency.** Worst case latency to transition a root hub Port Link State (PLS) from U1 to U0. Applies to all root hub ports. The following are permissible values: Value Description 00h Zero 01h Less than 1 us 02h Less than 2 us. ... 0Ah Less than 10 us. 0B-FFh Reserved

1A240010 HCCPARAMS Capability Parameters 01401198

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	XECP															
Type	RO															
Reset	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAXPSASIZE						SBD	PAE	NSS	LTC	LHRC	PIND	PPC	CSZ	BNC	AC64
Type	RO						RO	RO	RO	RO	RO	RO	RO	RO	RO	RO
Reset	0	0	0	1			0	1	1	0	0	1	1	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		XECP	**xHCI Extended Capabilities Pointer (xECP).** This field indicates the existence of a capabilities list. The value of this field indicates a relative offset, in 32-bit words, from Base to the beginning of the first extended capability. For example, using the offset of Base is 1000h and the xECP value of 0068h, we can calculated the following effective address of the first extended capability: 1000h + (0068h << 2) -> 1000h + 01A0h -> 11A0h

Bit(s)	Mnemonic	Name	Description
15:12		MAXPSASIZE	**Maximum Primary Stream Array Size (MaxPSASize).** This fields identifies the maximum size Primary Stream Array that the xHC supports. The Primary Stream Array size = 2MaxPSASize+1. Valid MaxPSASize values are 1 to 15.
9		SBD	**Secondary Bandwidth Domain Reporting (SBD).** This flag indicates whether the host controller implementation is capable of reporting Secondary Bandwidth Domain information. A '1' in this bit indicates that Secondary Bandwidth Domain reporting is supported. A '0' in this bit indicates that Secondary Bandwidth Domain reporting is not supported. Refer to Section 4.16.2 for more information on the use of this flag.
8		PAE	**Parse All Event Data (PAE). This flag indicates whether the host controller implementation** Parses all Event Data TRBs while advancing to the next TD after a short packet, or it skips all but the first Event Data TRB. A '1' in this bit indicates that all Event Data TRBs are parsed. A '0' in this bit indicates that only the first Event Data TRB is parsed (refer to section 4.10.1.1).
7		NSS	**No Secondary SID Support (NSS).** This flag indicates whether the host controller implementation supports Secondary Stream IDs. A '1' in this bit indicates that Secondary Stream ID decoding is not supported. A '0' in this bit indicates that Secondary Stream ID decoding is supported. (refer to Sections 4.12.2 and 6.2.3).
6		LTC	**Latency Tolerance Messaging Capability (LTC).** This flag indicates whether the host controller implementation supports Latency Tolerance Messaging (LTM). A '1' in this bit indicates that LTM is supported. A '0' in this bit indicates that LTM is not supported. Refer to section 4.13.1 for more information on LTM.
5		LHRC	**Light HC Reset Capability (LHRC).** This flag indicates whether the host controller implementation supports a Light Host Controller Reset.

Bit(s)	Mnemonic	Name	Description
			A '1' in this bit indicates that Light Host Controller Reset is supported. A '0' in this bit indicates that Light Host Controller Reset is not supported. The value of this flag affects the functionality of the Light Host Controller Reset (LHCRST) flag in the USBCMD register (refer to Section 5.4.1).
4		PIND	**Port Indicators (PIND).** This bit indicates whether the xHC root hub ports support port indicator control. When this bit is a '1', the port status and control registers include a read/ writeable field for controlling the state of the port indicator. Refer to Section 5.4.8 for definition of the Port Indicator Control field.
3		PPC	**Port Power Control (PPC).** This flag indicates whether the host controller implementation includes port power control. A '1' in this bit indicates the ports have port power switches. A '0' in this bit indicates the port do not have port power switches. The value of this flag affects the functionality of the PP flag in each port status and control register (refer to Section 5.4.8).
2		CSZ	**Context Size (CSZ).** If this bit is set to '1', then the xHC uses 64 byte Context data structures. If this bit is cleared to '0', then the xHC uses 32 byte Context data structures. Note: This flag does not apply to Stream Contexts.
1		BNC	**BW Negotiation Capability (BNC).** This flag identifies whether the xHC has implemented the Bandwidth Negotiation. Values for this flag have the following interpretation: Value Description 0 BW Negotiation not implemented 1 BW Negotiation implemented Refer to section 4.16 for more information on Bandwidth Negotiation.
0		AC64	**64-bit Addressing Capabilitya (AC64).** This flag documents the addressing range capability of this implementation. The value of this flag determines whether the xHC has implemented the high order 32 bits of 64 bit register and data structure pointer fields.

Bit(s)	Mnemonic	Name	Description
			Values for this flag have the following interpretation: Value Description 0 32-bit address memory pointers implemented 1 64-bit address memory pointers implemented If 32-bit address memory pointers are implemented, the xHC shall ignore the high order 32 bits of 64 bit data structure pointer fields, and system software shall ignore the high order 32 bits of 64 bit xHC registers.

1A240014 DBSOFF Doorbell Offset 00000800

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DBSOFF															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DBSOFF															
Type	RO															
Reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0		

Bit(s)	Mnemonic	Name	Description
31:2		DBSOFF	**Doorbell Array Offset.** Default = implementation dependent. This field defines the Dword offset of the Doorbell Array base address from the Base (i.e. the base address of the xHCI Capability register address space).

1A240018 RTSOFF Runtime Register Space Offset 00000600

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RTSOFF															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RTSOFF															
Type	RO															
Reset	0	0	0	0	0	1	1	0	0	0	0					

Bit(s)	Mnemonic	Name	Description
31:5		RTSOFF	**Runtime Register Space Offset.**

Bit(s)	Mnemonic	Name	Description
			Default = implementation dependent. This field defines the 32-byte offset of the xHCI Runtime Registers from the Base. i.e. Runtime Register Base Address = Base + Runtime Register Set Offset.

1A240020 USBCMD USB Command 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name					**EU3S**	**EWE**	**CRS**	**CSS**	**LHCRST**				**HSEE**	**INTE**	**HCRST**	**R_S**
Type					RW	RW	RO	RO	RO				RW	RW	RW	RW
Reset					0	0	0	0	0				0	0	0	0

Bit(s)	Mnemonic	Name	Description
11		EU3S	**Enable U3 MFINDEX Stop (EU3S).** When set to '1', the xHC may stop the MFINDEX counting action if all Root Hub ports are in the U3, Disconnected, Disabled, or Powered-off state. When cleared to '0' the xHC may stop the MFINDEX counting action if all Root Hub ports are in the Disconnected, Disabled, or Powered-off state.
10		EWE	**Enable Wrap Event (EWE).** Default = '0'. When set to '1', the xHC shall generate a MFINDEX Wrap Event every time the MFINDEX register transitions from 03FFFh to 0. When set to '0' no MFINDEX Wrap Events are generated. When this register is exposed by a Virtual Function (VF), the generation of MFINDEX Wrap Events to VFs must be emulated by the VMM.
9		CRS	**Controller Restore State (CRS).** When set to '1', Run/Stop = '0', and Save State = '1', the xHC shall perform a Restore State operation and restore

Bit(s)	Mnemonic	Name	Description
			its internal state. When set to '1' and Run/Stop = '1' or Save State = '0', or when set to '0', no Restore State operation shall be performed. This field always returns '0' when read. See the Restore State Status flag in the USBSTS register for information on Restore State completion. When this register is exposed by a Virtual Function (VF), this bit only controls restoring the state of the xHC instance presented by the selected VF.
8		CSS	**Controller Save State (CSS).** When written by software with '1' and Run/ Stop = '0', the xHC shall save any internal state that will be restored by a subsequent Restore State operation. When written by software with '1' and Run/Stop = '1', or written with '0', no Save State operation shall be performed. See the Save State Status flag in the USBSTS register for information on Save State completion. This field is set to '0' after initial power-up of the Auxiliary Power Well, by HCRST, or by LHCRST. When this register is exposed by a Virtual Function (VF), this bit only controls saving the state of the xHC instance presented by the selected VF
7		LHCRST	**Light Host Controller Reset (LHCRST).** Optional normative. If the Light HC Reset Capability (LHRC) bit in the HCCPARAMS register is '1', then this field allows the driver to reset the xHC without affecting the state of the ports. A system software read of this bit as '0' indicates the Light Host Controller Reset has completed and it is safe for software to re-initialize the xHC. A software read of this bit as a '1' indicates the Light Host Controller Reset has not yet completed. If not implemented, a read of this field will always return a '0'. All registers in the Auxiliary well will maintain the values that had been asserted prior to the Light Host Controller Reset. When this register is exposed by a Virtual Function (VF), this bit only generates a Light Reset to the xHC instance presented by the selected VF, e.g. Disable the VFs' device slots and set the associated VF Run bit to Stopped.
3		HSEE	**Host System Error Enable (HSEE).** When this bit is a '1', and the HSE bit in the USBSTS register is a '1', the xHC shall assert out-of-band error signaling to the host. The signaling is acknowledged by software clearing the HSE bit. When this register is

Bit(s)	Mnemonic	Name	Description
			exposed by a Virtual Function (VF), the effect of the assertion of this bit on the Physical Function (PF0) is determined by the VMM.
2		INTE	**Interrupter Enable (INTE).** This bit provides system software with a means of enabling or disabling the host system interrupts generated by Interrupters. When this bit is a '1', then Interrupter host system interrupt generation is allowed, e.g. the xHC shall issue an interrupt at the next interrupt threshold if the host system interrupt mechanism (e.g. MSI, MSI-X, etc.) is enabled. The interrupt is acknowledged by a host system interrupt specific mechanism. When this register is exposed by a Virtual Function (VF), this bit only enables the set of Interrupters assigned to the selected VF.
1		HCRST	**Host Controller Reset (HCRST).** This control bit is used by software to reset the host controller. The effects of this on Root Hub registers are similar to a Chip Hardware Reset. When software writes a '1' to this bit, the Host Controller resets its internal pipelines, timers, counters, state machines, etc. to their initial value. Any transaction currently in progress on USB is immediately terminated. A USB reset is not driven on downstream ports. PCI Configuration registers are not affected by this reset. All operational registers, including port registers and port state machines are set to their initial values. This bit is set to '0' by the Host Controller when the reset process is complete. Software cannot terminate the reset process early by writing a '0' to this bit. Software shall not set this bit to a '1' when the HCHalted bit in the USBSTS register is a '0'. Attempting to reset an actively running host controller will result in undefined behavior. When this register is exposed by a Virtual Function (VF), this bit only resets the xHC instance presented by the selected.
0		R_S	**Run/Stop (R/S).** '1' = Run. '0' = Stop. When set to a '1', the xHCI proceeds with execution of the schedule. The xHCI continues execution as long as this bit is set to a '1'. When this bit is set to '0', the xHCI completes the current and any actively pipelined transactions on the USB and then halts. The xHCI must halt within 16 microframes after

Bit(s)	Mnemonic	Name	Description
			software clears the Run/Stop bit. The HC Halted bit in the USBSTS register indicates when the xHC has finished its pending pipelined transactions and has entered the stopped state. Softwaremust not write a '1' to this field unless the xHC is in the Halted state (i.e. HCHalted in the USBSTS register is '1'). Doing so will yield undefined results. When this register is exposed by a Virtual Function (VF), this bit only controls the run state of the xHC instance presented by the selected VF.

1A240024 USBSTS USB Status 00000801

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				**HCE**	**CNR**	**SRE**	**RSS**	**SSS**				**PCD**	**EINT**	**HSE**		**HCH**
Type				RO	RU	RO	RO	RO				W1C	W1C	RO		RO
Reset				0	1	0	0	0				0	0	0		1

Bit(s)	Mnemonic	Name	Description
12		HCE	**Host Controller Error (HCE).** o' = No internal xHC error conditions exist and '1' = Internal xHC error condition. This flag shall be set to indicate that an internal error condition has been detected which requires software to reset and reinitialize the xHC.
11		CNR	**Controller Not Ready (CNR).** 'o' = Ready and '1' = Not Ready. Software shall not write any Doorbell or Operational register of the xHC, other than the USBSTS register, until CNR = 'o'. This flag is set by the xHC after a Chip Hardware Reset and cleared when the xHC is ready to begin accepting register writes. This flag shall remain cleared ('o') until the next Chip Hardware Reset.
10		SRE	**Save/Restore Error (SRE).**

Bit(s)	Mnemonic	Name	Description
			If an error occurs during a Save or Restore operation this bit shall be set to '1'. This bit shall be set to '0' when a Save or Restore operation is initiated or when written with '1'When this register is exposed by a Virtual Function (VF), the VMM determines the state of this bit as a function of the Save/Restore completion status for the selected VF.
9		RSS	**Restore State Status (RSS).** When the Controller Restore State flag in the USBCMD register is written with '1' this bit shall be set to '1' and remain 1 while the xHC restores its internal state. When the Restore State operation is complete, this bit shall be set to '0'. When this register is exposed by a Virtual Function (VF), the VMM determines the state of this bit as a function of the restoring the state for the selected VF.
8		SSS	**Save State Status (SSS).** When the Controller Save State flag in the USBCMD register is written with '1' this bit shall be set to '1' and remain 1 while the xHC saves its internal state. When the Save State operation is complete, this bit shall be set to '0'. When this register is exposed by a Virtual Function (VF), the VMM determines the state of this bit as a function of the saving the state for the selected VF.
4		PCD	**Port Change Detect (PCD).** The xHC sets this bit to a '1' when any port has a change bit transition from a '0' to a '1' or a Force Port Resume bit transition from a '0' to a '1' as a result of a resume detected on a suspended port. This bit is allowed to be maintained in the Auxiliary power well. Alternatively, it is also acceptable that on a D3 to D0 transition of the xHC, this bit is loaded with the OR of all of the PORTSC change bits (including: Force port resume, over-current change, enable/disable change and connect status change). This bit provides system software an efficient means of determining if there has been Root Hub port activity. When this register is exposed by a Virtual Function (VF), the VMM determines the state of this bit as a function of the Root Hub Ports associated with the Device Slots assigned to the selected VF.
3		EINT	**Event Interrupt (EINT).** The xHC sets this bit to '1' when the Interrupt Pending

Bit(s)	Mnemonic	Name	Description
			(IP) bit of any Interrupter is transitions from '0' to '1'. Software that uses EINT shall clear it prior to clearing any IP flags. A race condition will occur if software clears the IP flags then clears the EINT flag, and between the operations another IP '0' to '1' transition occurs. In this case the new IP transition will be lost. When this register is exposed by a Virtual Function (VF), this bit is the logical 'OR' of the IP bits for the Interrupters assigned to the selected VF. And it shall be cleared to '0' when all associated interrupter IP bits are cleared, i.e. all the VFs Interrupter Event Ring(s) are empty.
2		HSE	**Host System Error (HSE).** The xHC sets this bit to '1' when a serious error is detected, either internal to the xHC or during a host system access involving the xHC module. (In a PCI system, conditions that set this bit to '1' include PCI Parity error, PCI Master Abort, and PCI Target Abort.) When this error occurs, the xHC clears the Run/Stop bit in the USBCMD register to prevent further execution of the scheduled TDs. If the HSEE bit in the USBCMD register is a '1', the xHC shall also assert out-of-band error signaling to the host. When this register is exposed by a Virtual Function (VF), the assertion of this bit affects all VFs and reflects the Host System Error state of the Physical Function (PF0).
0		HCH	**HCHalted (HCH).** This bit is a '0' whenever the Run/Stop bit is a '1'. The xHC sets this bit to '1' after it has stopped executing as a result of the Run/Stop bit being set to '0', either by software or by the xHC hardware (e.g. internal error). When this register is exposed by a Virtual Function (VF), this bit only reflects the Halted state of the xHC instance presented by the selected VF.

1A240028 PAGESIZE Page Size 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PAGESIZE															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
15:0		PAGESIZE	**Page Size.** Default = Implementation defined. This field defines the page size supported by the xHC implementation. This xHC supports a page size of 2^(n+12) if bit n is Set. For example, if bit 0 is Set, the xHC supports 4k byte page sizes. For a Virtual Function, this register reflects the page size selected in the System Page Size field of the SR-IOV Extended Capability structure. For the Physical Function 0, this register reflects the implementation dependent default xHC page size. Various xHC resources reference PAGESIZE to describe their minimum alignment requirements. The maximum possible page size is 128M.

1A240034 DNCTRL Device Notification Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DNCTRL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		DNCTRL	**Notification Enable (No-N15).** When a Notification Enable bit is set, a Device Notification Event will be generated when a Device Notification Transaction Packet is received with the matching value in the Notification Type field. For

Bit(s)	Mnemonic	Name	Description
			example, setting N1 to '1' enables Device Notification Event generation if a Device Notification TP is received with its Notification Type field set to '1' (FUNCTION_WAKE), etc.

1A240038 CRCR1 Command Ring Control 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRP_Lo															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRP_Lo												CRR	CA	CS	RCS
Type	WO												RO	WO	WO	WO
Reset	0	0	0	0	0	0	0	0	0	0			0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:6		CRP_Lo	**Command Ring Pointer.** Default = '0'. This field defines high order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. Writes to this field are ignored when Command Ring Running (CRR) = '1'. If the CRCR is written while the Command Ring is stopped (CCR = '0'), the value of this field shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CCR = '0') then the Command Ring shall begin fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. Reading this field always returns '0'.
3		CRR	**Command Ring Running (CRR).** Default = 0. This flag is set to '1' if the Run/Stop (R/S) bit is '1' and the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. It is cleared to '0' when the Command Ring is "stopped" after writing a '1'to the Command Stop (CS) or Command Abort (CA) flags, or if the R/S bit is cleared to '0'.

Bit(s)	Mnemonic	Name	Description
2		CA	**Command Abort (CA).** Default = '0'. Writing a '1' to this bit shall immediately terminate the currently executing command, stop the Command Ring, and generate a Command Completion Event with the Completion Code set to Command Ring Stopped. Refer to section 4.6.1.2 for more information on aborting a command. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. Writes to this flag are ignored by the xHC if Command Ring Running (CRR) = '0'. Reading this bit always returns '0'.
1		CS	**Command Stop (CS).** Default = '0'. Writing a '1' to this bit shall stop the operation of the Command Ring after the completion of the currently executing command, and generate a Command Completion Event with the Completion Code set to Command Ring Stopped and the Command TRB Pointer set to the current value of the Command Ring Dequeue Pointer. Refer to section 4.6.1.1 for more information on stopping a command. The next write to the Host Controller Doorbell with DB Reason field set to Host Controller Command shall restart the Command Ring operation. Writes to this flag are ignored by the xHC if Command Ring Running (CRR) = '0'. Reading this bit shall always return '0'.
0		RCS	**Ring Cycle State (RCS).** This bit identifies the value of the xHC Consumer Cycle State (CCS) flag for the TRB referenced by the Command Ring Pointer. Refer to section 4.9.3 for more information. Writes to this flag are ignored if Command Ring Running (CRR) is '1'. If the CRCR is written while the Command Ring is stopped (CRR = '0'), then the value of this flag shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CRR = '0'), then the Command Ring will begin fetching Command TRBs using the current value of the internal Command Ring CCS flag.

Bit(s)	Mnemonic	Name	Description
			Reading this flag always returns '0'.

1A24003C CRCR2 Command Ring Control 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CRP_Hi															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	CRP_Hi															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		CRP_Hi	**Command Ring Pointer.** Default = '0'. This field defines high order bits of the initial value of the 64-bit Command Ring Dequeue Pointer. Writes to this field are ignored when Command Ring Running (CRR) = '1'. If the CRCR is written while the Command Ring is stopped (CCR = '0'), the value of this field shall be used to fetch the first Command TRB the next time the Host Controller Doorbell register is written with the DB Reason field set to Host Controller Command. If the CRCR is not written while the Command Ring is stopped (CCR = '0') then the Command Ring shall begin fetching Command TRBs at the current value of the internal xHC Command Ring Dequeue Pointer. Reading this field always returns '0'.

1A240050 DCBAAP_LO Device Context Base Address Array Pointer Lo 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DCBAAP_Lo															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCBAAP_Lo															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0						

Bit(s)	Mnemonic	Name	Description
31:6		DCBAAP_Lo	**Device Context Base Address Array Pointer.** Default = '0'. This field defines high order bits of the 64-bit base address of the Device Context Pointer Array table. A table of address pointers that reference Device Context structures for the devices attached to the host.

1A240054 DCBAAP_HI Device Context Base Address Array Pointer High 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DCBAAP_Hi															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DCBAAP_Hi															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		DCBAAP_Hi	**Device Context Base Address Array Pointer.** This field defines [63:32] bits of the 64-bit base address of the Device Context Pointer Array table. A table of address pointers that reference Device Context structures for the devices attached to the host.

1A240058 CONFIG Configure 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									MaxSlotsEn							
Type									RW							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
7:0		MaxSlotsEn	**Max Device Slots Enabled (MaxSlotsEn).** Default = '0'. This field specifies the maximum number of enabled Device Slots. Valid values are in the range of 0 to MaxSlots. Enabled Devices Slots are allocated contiguously. e.g. A value of 16 specifies that Device Slots 1 to 16 are active. A value of '0' disables all Device Slots. A disabled Device Slot shall not respond to Doorbell Register references. This field shall not be modified if the xHC is running (Run/Stop (R/S) = '1').

1A240500 SUPP_PTCL_REG1 xHCI Supported Protocol Capability 1 Register 03000402

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Revision_Major								Revision_Minor							
Type	RO								RO							
Reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Next_Capability_Pointer								Capability_ID							
Type	RO								RO							
Reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:24		Revision_Major	**Major Revision.** Major Specification Release Number in Binary-Coded Decimal (i.e., version 3.x is 03h). This field identifies the major release number component of the specification with which the xHC is compliant.
23:16		Revision_Minor	**Minor Revision.** Minor Specification Release Number in Binary-Coded Decimal (i.e., version x.10 is 10h). This field identifies the minor release number component of the specification with which the xHC is compliant.
15:8		Next_Capability_Pointer	**Next Capability Pointer.** This field indicates the location of the next capability with respect to the effective address of this capability. Refer to Table 137 for more information on this field.

Bit(s)	Mnemonic	Name	Description
7:0		Capability_ID	**Capability ID.** Refer to Table 138 for the value that identifies the capability as Supported Protocol.

1A240504 SUPP_PTCL_REG2 xHCI Supported Protocol Capability 2 Register 20425355

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Name_String															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Name_String															
Type	RO															
Reset	0	1	0	1	0	0	1	1	0	1	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
31:0		Name_String	**Name String.** This field is a mnemonic name string that references the specification with which the xHC is compliant. Four ASCII characters may be defined. Allowed characters are: alphanumeric, space, and underscore. Alpha characters are case sensitive. Refer to Table 145 for defined values.

1A240508 SUPP_PTCL_REG3 xHCI Supported Protocol Capability 3 Register 00000201

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														IHI	HSO	L1C
Type														RO	RO	RO
Reset														0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Compatible_Port_Count								Compatible_Port_Offset							
Type	RO								RO							

Reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1

Bit(s)	Mnemonic	Name	Description
18		IHI	**Integrated Hub Implemented (IHI).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is cleared to '0', the Root Hub to External xHC port mapping adheres to the default mapping described in section 4.24.2.1. If this bit is set to '1', the Root Hub to External xHC port mapping does not adhere to the default mapping described in section 4.24.2.1, and an ACPI or other mechanism is required to define the mapping.
17		HSO	**High-speed Only (HSO).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is set to '0', the USB2 ports described by this capability are Low-, Full-, and High-speed capable. If this bit is set to '1', the USB2 ports described by this capability are High-speed only, e.g. the ports don't support Low- or Full-speed operation. High-speed only implementations may introduce a "Tier mismatch", refer to section 4.24.2 for more information.
16		L1C	**L1 Capability (L1C).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is set to '1' the xHC supports the USB2 Link Power Management L1 (Sleep) state and the associated USB2 protocol fields as defined in the PORTSC and USB2 PORTPMSC registers are valid, specifically USB2 protocol functionality of the PLS and PLC fields in the PORTSC register, and the fields of the USB2 PORTPMSC register. Note that software is prohibited from using the PLS field initiate a transition to an L1 state or using the USB2 PORTPMSC fields unless this bit is set to '1'.
15:8		Compatible_Port_Count	**Compatible Port Count.** This field identifies the number of consecutive Root Hub Ports

Bit(s)	Mnemonic	Name	Description
			(starting at the Compatible Port Offset) that support this protocol. Valid values are 1 to MaxPorts.
7:0		Compatible_Port_Offset	**Compatible Port Offset.** This field specifies the starting Port Number of Root Hub Ports that support this protocol. Valid values are '1' to MaxPorts.

1A240510 SUPP_PTCL_REG4 xHCI Supported Protocol Capability 1 Register 02000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Revision_Major								Revision_Minor							
Type	RO								RO							
Reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Next_Capability_Pointer								Capability_ID							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Bit(s)	Mnemonic	Name	Description
31:24		Revision_Major	**Major Revision.** Major Specification Release Number in Binary-Coded Decimal (i.e., version 3.x is 03h). This field identifies the major release number component of the specification with which the xHC is compliant.
23:16		Revision_Minor	**Minor Revision.** Minor Specification Release Number in Binary-Coded Decimal (i.e., version x.10 is 10h). This field identifies the minor release number component of the specification with which the xHC is compliant.
15:8		Next_Capability_Pointer	**Next Capability Pointer.** This field indicates the location of the next capability with respect to the effective address of this capability. Refer to Table 137 for more information on this field.

Bit(s)	Mnemonic	Name	Description
7:0		Capability_ID	**Capability ID.** Refer to Table 138 for the value that identifies the capability as Supported Protocol.

1A240514 SUPP_PTCL_REG5 xHCI Supported Protocol Capability 2 Register 20425355

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Name_String															
Type	RO															
Reset	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Name_String															
Type	RO															
Reset	0	1	0	1	0	0	1	1	0	1	0	1	0	1	0	1

Bit(s)	Mnemonic	Name	Description
31:0		Name_String	**Name String.** This field is a mnemonic name string that references the specification with which the xHC is compliant. Four ASCII characters may be defined. Allowed characters are: alphanumeric, space, and underscore. Alpha characters are case sensitive. Refer to Table 145 for defined values.

1A240518 SUPP_PTCL_REG6 xHCI Supported Protocol Capability 3 Register 00010503

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name														IHI	HSO	L1C
Type														RO	RO	RO
Reset														0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Compatible_Port_Count								Compatible_Port_Offset							
Type	RO								RO							

Reset	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	1

Bit(s)	Mnemonic	Name	Description
18		IHI	**Integrated Hub Implemented (IHI).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is cleared to '0', the Root Hub to External xHC port mapping adheres to the default mapping described in section 4.24.2.1. If this bit is set to '1', the Root Hub to External xHC port mapping does not adhere to the default mapping described in section 4.24.2.1, and an ACPI or other mechanism is required to define the mapping.
17		HSO	**High-speed Only (HSO).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is set to '0', the USB2 ports described by this capability are Low-, Full-, and High-speed capable. If this bit is set to '1', the USB2 ports described by this capability are High-speed only, e.g. the ports don't support Low- or Full-speed operation. High-speed only implementations may introduce a "Tier mismatch", refer to section 4.24.2 for more information.
16		L1C	**L1 Capability (L1C).** Default = Implementation dependent. This field only applies to the USB2 protocol. If this bit is set to '1' the xHC supports the USB2 Link Power Management L1 (Sleep) state and the associated USB2 protocol fields as defined in the PORTSC and USB2 PORTPMSC registers are valid, specifically USB2 protocol functionality of the PLS and PLC fields in the PORTSC register, and the fields of the USB2 PORTPMSC register. Note that software is prohibited from using the PLS field initiate a transition to an L1 state or using the USB2 PORTPMSC fields unless this bit is set to '1'.
15:8		Compatible_Port_Count	**Compatible Port Count.** This field identifies the number of consecutive Root Hub Ports

Bit(s)	Mnemonic	Name	Description
			(starting at the Compatible Port Offset) that support this protocol. Valid values are 1 to MaxPorts.
7:0		Compatible_Port_Offset	**Compatible Port Offset.** This field specifies the starting Port Number of Root Hub Ports that support this protocol. Valid values are '1' to MaxPorts.

1A240600 MFINDEX Microframe Index 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			MICROFRAME_INDEX													
Type			RO													
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
13:0		MICROFRAME_INDEX	**Microframe Index.** The value in this register increments at the end of each microframe (e.g. 125us.). Bits [13:3] may be used to determine the current 1ms. Frame Index.

1A240620 IMAN Interrupter Management 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Na															IE	IP

me																
Type															RW	W1C
Reset															0	0

Bit(s)	Mnemonic	Name	Description
1		IE	**Interrupt Enable (IE).** Default = '0'. This flag specifies whether the Interrupter is capable of generating an interrupt. When this bit and the IP bit are set ('1'), the Interrupter shall generate an interrupt when the Interrupter Moderation Counter reaches '0'. If this bit is '0', then the Interrupter is prohibited from generating interrupts.
0		IP	**Interrupt Pending (IP).** Default = '0'. This flag represents the current state of the Interrupter. If IP = '1', an interrupt is pending for this Interrupter. This flag is set to '1' when IE = '1', the IMODC Interrupt Moderation Counter field = '0' the Event Ring associated with the Interrupter is not empty, and EHB = '0'. A '0' value indicates that no interrupt is pending for the Interrupter. If MSI interrupts are enabled, this flag shall be cleared automatically when the PCI Dword write generated by the Interrupt assertion is complete. If PCI Pin Interrupts are enabled, this flag shall be cleared by software.

1A240624 IMOD Interrupter Moderation 00000FA0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IMODC															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IMODI															
Type	RW															
Reset	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		IMODC	**Interrupt Moderation Counter (IMODC).** Default = undefined. Down counter. Loaded with Interval value whenever IP is cleared to '0', counts down to '0', and stops. The associated interrupt shall be signaled whenever this counter is '0', the Event Ring is not empty, the IE and IP flags = '1', and EHB = '0'.

Bit(s)	Mnemonic	Name	Description
			This counter may be directly written by software at any time to alter the interrupt rate.
15:0		IMODI	**Interrupt Moderation Interval (IMODI).** Default = '4000'. Minimum inter-interrupt interval. The interval is specified inunit of 256 internal xHC clock cycles. A value of '0' disables interrupt throttling logic and interrupts shall be generated immediately if IP = '0', EHB = '0', and the Event Ring is not empty.

1A240628 ERSTSZ Event Ring Segment Table Size 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERSTSZ															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		ERSTSZ	**Event Ring Segment Table Size.** Default = '0'. This field identifies the number of valid Event Ring Segment Table entries in the Event Ring Segment Table pointed to by the Event Ring Segment Table Base Address register. The maximum value supported by an xHC implementation for this register is defined by the ERST Max field in the HSCPARAMS2 register (5.3.4). For Secondary Interrupters: Writing a value of '0' to this field disables the Event Ring. Any events targeted at this Event Ring when it is disabled shall result in undefined behavior of the Event Ring. For the Primary Interrupter: Writing a value of '0' to this field shall result in undefined behavior of the Event Ring. The Primary Event Ring cannot be disabled.

1A240630 ERSTBA_LO Event Ring Segment Table Base Address Lo 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ERSTBA_LO															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERSTBA_LO															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0				

Bit(s)	Mnemonic	Name	Description
31:4		ERSTBA_LO	**Event Ring Segment Table Base Address Register.** Default = '0'. This field defines the high order bits of the start address of the Event Ring Segment Table. Writing this register sets the Event Ring State Machine:EREP Advancement to the Start state. This field shall not be modified if HCHalted (HCH) = '0'.

1A240634 ERSTBA_HI Event Ring Segment Table Base Address Hi 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ERSTBA_HI															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERSTBA_HI															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		ERSTBA_HI	**Event Ring Segment Table Base Address Register.** Default = '0'. This field defines the high order bits of the start address of the Event Ring Segment Table. Writing this register sets the Event Ring State Machine:EREP Advancement to the Start state. This field shall not be modified if HCHalted (HCH) = '0'.

1A240638 ERDP_LO Event Ring Segment Table Base Address Lo 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ERDP_LO															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERDP_LO												EHB	DESI		
Type	RW												W1C	RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:4		ERDP_LO	**Event Ring Dequeue Pointer.** Default = '0'. This field defines the high order bits of the 64- bit address of the current Event Ring Dequeue Pointer.
3		EHB	**Event Handler Busy (EHB).** Default = '0'. This flag shall be set to '1' when the IP bit is set to '1' and cleared to '0' by software when the Dequeue Pointer register is written. Refer to section 4.17.2 for more information
2:0		DESI	**Dequeue ERST Segment Index (DESI).** Default = '0'. This field may be used by the xHC to accelerate checking the Event Ring full condition. This field is written with the low order 3 bits of the offset of the ERST entry which defines the Event Ring segment that Event Ring Dequeue Pointer resides in.

1A24063C ERDP_HI Event Ring Segment Table Base Address Hi 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ERDP_HI															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ERDP_HI															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		ERDP_HI	**Event Ring Dequeue Pointer.** Default = '0'. This field defines the high order bits of the 64- bit address of the current Event Ring Dequeue

Bit(s)	Mnemonic	Name	Description
			Pointer.

1A240800 HOST_CMD_DB Host Controller Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	HOST_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									HOST_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		HOST_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		HOST_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update

Bit(s)	Mnemonic	Name	Description
			31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240804 DEVICE1_DB Device 1 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE1_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE1_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE1_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE1_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other

Bit(s)	Mnemonic	Name	Description
			Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240808 DEVICE2_DB Device 2 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE2_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE2_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE2_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the

Bit(s)	Mnemonic	Name	Description
			doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE2_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A24080C DEVICE3_DB Device 3 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE3_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE3_DB_TARGET							
Type									WO							

Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE3_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE3_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240810 DEVICE4_DB Device 4 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE4_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE4_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE4_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE4_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition

Bit(s)	Mnemonic	Name	Description
			0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240814 DEVICE5_DB Device 5 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE5_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE5_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE5_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE5_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update

Bit(s)	Mnemonic	Name	Description
			2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240818 DEVICE6_DB Device 6 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE6_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE6_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE6_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.

Bit(s)	Mnemonic	Name	Description
7:0		DEVICE6_DB_TARGET	**DB Target - RW. Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers.** Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A24081C DEVICE7_DB Device 7 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE7_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE7_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE7_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context

Bit(s)	Mnemonic	Name	Description
			Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE7_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240820 DEVICE8_DB Device 8 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE8_DB_STREAM_ID															

Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE8_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE8_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE8_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated

Bit(s)	Mnemonic	Name	Description
			as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240824 DEVICE9_DB Device 9 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE9_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE9_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE9_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE9_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update

Bit(s)	Mnemonic	Name	Description
			... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240828 DEVICE10_DB Device 10 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE10_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE10_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE10_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE10_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell.

Bit(s)	Mnemonic	Name	Description
			Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A24082C DEVICE11_DB Device 11 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE11_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE11_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE11_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid.

Bit(s)	Mnemonic	Name	Description
			If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE11_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240830 DEVICE12_DB Device 12 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE12_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name									DEVICE12_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE12_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE12_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

Bit(s)	Mnemonic	Name	Description

1A240834 DEVICE13_DB Device 13 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE13_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE13_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE13_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE13_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update

Bit(s)	Mnemonic	Name	Description
			32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240838 DEVICE14_DB Device 14 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE14_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE14_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE14_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored. This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE14_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers.

Bit(s)	Mnemonic	Name	Description
			Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A24083C DEVICE15_DB Device 15 Doorbell Registers 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	DEVICE15_DB_STREAM_ID															
Type	WO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									DEVICE15_DB_TARGET							
Type									WO							
Reset									0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:16		DEVICE15_DB_STREAM_ID	**DB Stream ID.** Doorbell Stream ID. If the endpoint of a Device Context Doorbell defines Streams, then this field shall be used to identify which Stream of the endpoint the doorbell reference is targeting. System software is responsible for ensuring that the value written to this field is valid. If the endpoint does not define Streams (MaxPStreams = 0) and a non-'0' value is written to this field, the doorbell reference shall be ignored.

Bit(s)	Mnemonic	Name	Description
			This field only applies to Device Context Doorbells and shall be cleared to '0' for Host Controller Commands. This field returns '0' when read.
7:0		DEVICE15_DB_TARGET	**DB Target.** Doorbell Target. This field defines the target of the doorbell reference. The table below defines the xHC notification that is generated by ringing the doorbell. Note that Doorbell Register 0 is dedicated to Command Ring and decodes this field differently than the other Doorbell Registers. Device Context Doorbells (1-255) Value Definition 0 Reserved 1 Control EP 0 Enqueue Pointer Update 2 EP 1 OUT Enqueue Pointer Update 3 EP 1 IN Enqueue Pointer Update 4 EP 2 OUT Enqueue Pointer Update 5 EP 2 IN Enqueue Pointer Update ... 30 EP 15 OUT Enqueue Pointer Update 31 EP 15 IN Enqueue Pointer Update 32:247 Reserved 248:255 Vendor Defined Host Controller Doorbell (0) Value Definition 0 Host Controller Command 1:247 Reserved 248:255 Vendor Defined This field returns '0' when read and should be treated as "undefined" by software. When the Host Controller Doorbell (0) is written, the DB Stream ID field shall be cleared to '0'.

1A240900 HSRAM_DBGCTL Host SRAM Debug Control Register 00000002

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Na															**SRA**	**SRA**

me															M_DBG_WP	M_DBG_EN
Type															RW	RW
Reset															1	0

Bit(s)	Mnemonic	Name	Description
1		SRAM_DBG_WP	**SRAM Write Protect for Debug Mode** 0: allow SRAM Debug Write 1: prohibit SRAM Debug Write
0		SRAM_DBG_EN	**SRAM Debug Access Enable** 0: SRAM Debug Access is ignored 1: SRAM Debug Access is allowed

1A240904 HSRAM_DBGMODE Host SRAM Debug Mode Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SRAM_DBG_MODE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SRAM_DBG_MODE															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		SRAM_DBG_MODE	**Debug mode for host SRAM** Each bit indicates the SRAM debug mode of corresponding SRAM. For each bit, 0: SRAM in normal Mode. HW access is allowed. 1: SRAM in Debug Mode. HW access is prevented. bit[0]: ER SRAM (32x128) bit[1]: CMD SRAM (15x30) bit[2]: EP MAP (480x7) bit[3]: EP Slot (15x41) bit[4]: EP Static (64x75) bit[5]: EP Dynamic (64x51) bit[6]: TRBQ (64x114) bit[7]: USB2_DMA (64x128) bit[8]: SCH3 IN SRAM (256x128) bit[9]: SCH3 OUT SRAM (256x128)

Bit(s)	Mnemonic	Name	Description
			bit[10]: SCH2_0 (256x128) bit[11]: SCH2_1 (256x128) bit[12]: SCH2_2 (256x128) bit[13]: SCH2_3 (256x128) bit[14]: SCH2_4 (256x128) Note: More then one bit can be set to 1'b1 in HSRAM_DBGMODE, but only one of them in HSRAM_DBGSEL can be set to 1'b1.

1A240908 HSRAM_DBGSEL Host SRAM Debug Select Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SRAM_DBG_SEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SRAM_DBG_SEL															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		SRAM_DBG_SEL	**Debug Select for Host SRAM** Each bit indicates whether the access through HSRAM_DBGADR and HSRAM_DBGDR are targeted to related SRAM. Note: at most one bit in HSRAM_DBGSEL can be set to 1'b1 Note: when a specific bit is set to 1'b1, the related bit in HSRAM_DBGMODE shall be set to 1'b1 too.

1A24090C HSRAM_DBGADR Host SRAM Debug Address Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SRAM_DBG_ADDR														SRAM_DBG_DWSEL	
Type	RW														RW	

e																
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:2		SRAM_DBG_ADDR	**SRAM Address (16-byte access) for Debug Mode** This is equivalent to SRAM Address in HW because the SRAM's width is limit to 128-bit. Because the Host Slave interface is 32-bit, the SRAM_DBG_DWSEL[1:0] decide which 4-byte is read/write through SRAM_DBG_DATA[31:0]
1:0		SRAM_DBG_DWSEL	**SRAM DW Select (4-byte access) for Debug Mode** This 2-bit selects which 4-byte out of the 16-byte indicated by SRAM_DBG_ADDR[13:0] is read/write through SRAM_DBG_DATA[31:0] Note: The constraints on SRAM_DBG_DWSEL[1:0] according to the actual SRAM width in HW shall be followed: - For SRAM with 0 < width <= 32, the valid value of SRAM_DBG_DWSEL[1:0] = 0 - For SRAM with 32 < width <= 64, the valid value of SRAM_DBG_DWSEL[1:0] = 0, 1 - For SRAM with 64 < width <= 96, the valid value of SRAM_DBG_DWSEL[1:0] = 0, 1, 2 - For SRAM with 96 < width <= 128, the valid value of SRAM_DBG_DWSEL[1:0] = 0, 1, 2, 3

1A240910 HSRAM_DBGDR Host SRAM Debug Data Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SRAM_DBG_DATA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SRAM_DBG_DATA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		SRAM_DBG_DATA	**SRAM Debug Data Register** When Read, need to program HSRAM_DBGADR first and then read HSRAM_DBGDR and the selected SRAM's related 4-byte data is returned.

Bit(s)	Mnemonic	Name	Description
			When Write, need to program HSRAM_DBGADR first then read HSRAM_DBGDR and the data is stored to target location at specified address of the selected SRAM

1A240920 HSRAM_DELSEL_0 Host SRAM Delay Select 0 0000AAAA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DELSEL_U2DMA_0		DELSEL_TRBQ		DELSEL_EP_DYN		DELSEL_EP_STA		DELSEL_EP_SLT		DELSEL_EP_MAP		DELSEL_CMD		DELSEL_ER	
Type	RW		RW		RW		RW		RW		RW		RW		RW	
Reset	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
15:14		DELSEL_U2DMA_0	**DELSEL for USB2 DMA SRAM**
13:12		DELSEL_TRBQ	**DELSEL for CMD SRAM**
11:10		DELSEL_EP_DYN	**DELSEL for EP Dynamic SRAM**
9:8		DELSEL_EP_STA	**DELSEL for EP Static SRAM**
7:6		DELSEL_EP_SLT	**DELSEL for EP SLOT SRAM**
5:4		DELSEL_EP_MAP	**DELSEL for EP MAP SRAM**
3:2		DELSEL_CMD	**DELSEL for CMD SRAM SRAM**
1:0		DELSEL_ER	**DELSEL for ER SRAM**

1A240924 HSRAM_DELSEL_1 Host SRAM Delay Select 1 00002AAA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			DELSEL_SCH2_4		DELSEL_SCH2_3		DELSEL_SCH2_2		DELSEL_SCH2_1		DELSEL_SCH2_0		DELSEL_SCH3_OUT		DELSEL_SCH3_IN	
Type			RW		RW		RW		RW		RW		RW		RW	
Reset			1	0	1	0	1	0	1	0	1	0	1	0	1	0

Bit(s)	Mnemonic	Name	Description
13:12		DELSEL_SCH2_4	**DELSEL for USB2 Port 4 Scheduler SRAM**
11:10		DELSEL_SCH2_3	**DELSEL for USB2 Port 3 Scheduler SRAM**
9:8		DELSEL_SCH2_2	**DELSEL for USB2 Port 2 Scheduler SRAM**
7:6		DELSEL_SCH2_1	**DELSEL for USB2 Port 1 Scheduler SRAM**
5:4		DELSEL_SCH2_0	**DELSEL for USB2 Port 0 Scheduler SRAM**
3:2		DELSEL_SCH3_OUT	**DELSEL for USB3 OUT Scheduler SRAM**
1:0		DELSEL_SCH3_IN	**DELSEL for USB3 IN Scheduler SRAM**

1A240930 LS_EOF Low Speed EOF Start Offset 070E0128

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						LS_EOF_UFRAME							LS_EOF_BANK			
Type						RW							RW			
Reset						1	1	1					1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								LS_EOF_OFFSET								

e																
Type								RW								
Reset								1	0	0	1	0	1	0	0	0

Bit(s)	Mnemonic	Name	Description
26:24		LS_EOF_UFRAME	**Low Speed EOF uFrame** The LS is using 1ms, 8 micro frame, as frame period. This register set the micro number that LSEOF asserts
19:16		LS_EOF_BANK	**Low Speed EOF Bank** One micro frame, 125 us, is divided into 15 banks which is 500 cycles of 60 MHz for each. This register set the bank that LSEOF asserts. The valid value can be 0 to 14.
8:0		LS_EOF_OFFSET	**Low Speed EOF Offset** This register set the offset within a bank that LSEOF asserts. The valid value is 0~499.

1A240934 FS_EOF Full Speed EOF Start Offset 07070134

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name						FS_EOF_UFRAME							FS_EOF_BANK			
Type						RW							RW			
Reset						1	1	1					0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								FS_EOF_OFFSET								
Type								RW								
Reset								1	0	0	1	1	0	1	0	0

Bit(s)	Mnemonic	Name	Description
26:24		FS_EOF_UFRAME	**Full Speed EOF uFrame** The FS is using 1ms, 8 micro frame, as frame period. This register set the micro number that FSEOF asserts
19:16		FS_EOF_BANK	**Full Speed EOF Bank** One micro frame, 125 us, is divided into 15 banks which is 500 cycles of 60 MHz for each. This register set the bank that FSEOF asserts. The valid value can be 0 to

Bit(s)	Mnemonic	Name	Description
			14.
8:0		FS_EOF_OFFSET	**Full Speed EOF Offset** This register set the offset within a bank that FSEOF asserts. The valid value is 0~499.

1A240938 SYNC_HS_EOF Synchronous High Speed EOF Start Offset 000201F3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													SYNC_HS_EOF_BANK			
Type													RW			
Reset													0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								SYNC_HS_EOF_OFFSET								
Type								RW								
Reset								1	1	1	1	1	0	0	1	1

Bit(s)	Mnemonic	Name	Description
19:16		SYNC_HS_EOF_BANK	**Synchronous High Speed EOF Bank** One micro frame, 125 us, is divided into 15 banks which is 500 cycles of 60 MHz for each. This register set the bank that Synchronous HSEOF asserts. The valid value can be 0 to 14.
8:0		SYNC_HS_EOF_OFFSET	**Synchronous High Speed EOF Offset** This register set the offset within a bank that Synchronous HSEOF asserts. The valid value is 0~499.

1A24093C SS_EOF Super Speed EOF Start Offset 00000096

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													SS_EOF_BANK			
Type													RW			
Rese													0	0	0	0

t																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								SS_EOF_OFFSET								
Type								RW								
Reset								0	1	0	0	1	0	1	1	0

Bit(s)	Mnemonic	Name	Description
19:16		SS_EOF_BANK	**Super Speed EOF Bank** One micro frame, 125 us, is divided into 15 banks which is 500 cycles of 60 MHz for each. This register set the bank that SSEOF asserts. The valid value can be 0 to 14.
8:0		SS_EOF_OFFSET	**Super Speed EOF Offset** This register set the offset within a bank that SSEOF asserts. The valid value is 0~499.

1A240940 SOF_OFFSET SOF Offset 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SOF_U2_PORT5				SOF_U2_PORT4				SOF_U2_PORT3				SOF_U2_PORT2			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SOF_U2_PORT1				SOF_U2_PORT0				SOF_U3_PORT1				SOF_U3_PORT0			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:28		SOF_U2_PORT5	**USB2 PORT 5 SOF Bank** This register configures the offset bank for USB2 Port 5 SOF. The valid value can be 0 to 14.
27:24		SOF_U2_PORT4	**USB2 PORT 4 SOF Bank** This register configures the offset bank for USB2 Port 4 SOF. The valid value can be 0 to 14.
23:20		SOF_U2_PORT3	**USB2 PORT 3 SOF Bank** This register configures the offset bank for USB2 Port 3 SOF. The valid value can be 0 to 14.

Bit(s)	Mnemonic	Name	Description
19:16		SOF_U2_PORT2	**USB2 PORT 2 SOF Bank** This register configures the offset bank for USB2 Port 2 SOF. The valid value can be 0 to 14.
15:12		SOF_U2_PORT1	**USB2 PORT 1 SOF Bank** This register configures the offset bank for USB2 Port 1 SOF. The valid value can be 0 to 14.
11:8		SOF_U2_PORT0	**USB2 PORT 0 SOF Bank** This register configures the offset bank for USB2 Port 0 SOF. The valid value can be 0 to 14.
7:4		SOF_U3_PORT1	**USB3 PORT 1 SOF Bank** This register configures the offset bank for USB3 Port 1 SOF. The valid value can be 0 to 14.
3:0		SOF_U3_PORT0	**USB3 PORT 0 SOF Bank** This register configures the offset bank for USB3 Port 0 SOF. The valid value can be 0 to 14.

1A240944 HFCNTR_CFG Host Frame Counter Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																FCNTR_DIS
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0		FCNTR_DIS	**Frame Counter Disable** When this bit is set, the frame counter will stop counting.

Bit(s)	Mnemonic	Name	Description

1A240948 XACT3_CFG Super Speed Transaction Configuration 01010020

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								XACT3_PM_TRANS_EN							XACT3_ISO_OUT_TX_ZLP_DIS	XACT3_ISO_IN_CRC_CHK_DIS
Type								RW							RW	RW
Reset								1							0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									XACT3_TMOUT							
Type									RW							
Reset									0	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
24		XACT3_PM_TRANS_EN	**Xactor3 Power Mode transfer enable** When set to 1'b1, if there were residual transactions that not transfer to device in power mode, the Xactor3 will let the link go back to U0 and transfer them. When set to 1'b0, the Xactor3 will not transfer the residual transaction when in power mode.
17		XACT3_ISO_OUT_TX_ZLP_DIS	**Xactor3 Isochronous OUT TX ZLP Disable** When set to 1'b1, the Xactor3 does not TX ZLP for Isochronous OUT endpont. When set to 1'b0, the Xactor3 can TX ZLP for Isochronous OUT endpont.
16		XACT3_ISO_IN_CRC_CHK_DIS	**Xactor3 Isochronous IN CRC Check Disable** When set to 1'b1, the CRC Error of Isoch IN DP is ignored When set to 1'b0, the CRC Error of Isoch IN DP is reported

Bit(s)	Mnemonic	Name	Description
7:0		XACT3_TMOUT	**USB3 Transactor Timeout Value** This register controls the timeout value of XACT3. Unit in us.

1A24094C XACT2_CFG USB2 Transaction Configuration 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name														**XACT2_ISO_OUT_TX_ZLP_DIS**	**XACT2_SPLIT_ISO_IN_CRC_CHK_DIS**	**XACT2_NON_SPLIT_ISO_IN_CRC_CHK_DIS**
Type														RW	RW	RW
Reset														0	0	1

Bit(s)	Mnemonic	Name	Description
2		XACT2_ISO_OUT_TX_ZLP_DIS	**Xactor2 Isochronous OUT TX ZLP Disable** When set to 1'b1, the Xactor3 does not TX ZLP for Isochronous OUT endpont. When set to 1'b0, the Xactor3 can TX ZLP for Isochronous OUT endpont.
1		XACT2_SPLIT_ISO_IN_CRC_CHK_DIS	**Xactor2 non Split Isochronous IN CRC Check Disable** When set to 1'b1, the CRC Error of Isoch IN DP is ignored When set to 1'b0, the CRC Error of Isoch IN DP is reported

Bit(s)	Mnemonic	Name	Description
0		XACT2_NON_SPLIT_ISO_IN_CRC_CHK_DIS	**Xactor2 non Split Isochronous IN CRC Check Disable** When set to 1'b1, the CRC Error of Isoch IN DP is ignored When set to 1'b0, the CRC Error of Isoch IN DP is reported

1A240950 HDMA_CFG Host DMA Configuration 01020200

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name								DYN_DRAM_CG_EN		DMAU2_LIMITER			DMAU2_BURST		DMAU2_BUFFERABLE	DMAU2_FAKE
Type								RW		RW			RW		RW	RW
Reset								1		0	0	0	0	0	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMAW_LAST_NONBUF	DMAW_LIMITER			DMAW_BURST		DMAW_BUFFERABLE	DMAW_FAKE		DMAR_LIMITER			DMAR_BURST			DMAR_FAKE
Type	RW	RW			RW		RW	RW		RW			RW			RW
Reset	0	0	0	0	0	0	1	0		0	0	0	0	0		0

Bit(s)	Mnemonic	Name	Description
24		DYN_DRAM_CG_EN	**Dynamic DRAM Clock Gating Enable** 0: Dynamic clock gating is disabled. The clock gating on dram_ck is controlled by all root hub ports' link state 1: Dynamic clock gating is enabled. The clock gating is controll by the related dma requests
22:20		DMAU2_LIMITER	**DMA U2 Limiter Configuration** This register configures the number of DRAM cycle between two DMA request. 0: at least 1T

Bit(s)	Mnemonic	Name	Description
			1: at least 2T 2: at least 4T 3: at least 8T 4: at least 16T 5: at least 32T 6: at least 64T 7: at least 128T
19:18		DMAU2_BURST	**DMA U2 Burst Size Configuraiton** This register configures the maximum burst size per DMA request to DRAM bus. 0: 1024 bytes 1: 512 bytes 2: 256 bytes 3: 128 bytes
17		DMAU2_BUFFERABLE	**DMA U2 Bufferable Configuration** When this bit is set, the DMA U2 write operation is bufferable on target DMA bus
16		DMAU2_FAKE	**DMA U2 Fake Configuration** When this bit is set, no data will be moved but DMA returns ack immediately.
15		DMAW_LAST_NONBUF	**DMA Write Last burst Nonbufferable** 0: fully controled by DMAW_BUFFERABLE 1: force nonbufferable when last burst.
14:12		DMAW_LIMITER	**DMA Write Limiter Configuration** This register configures the number of DRAM cycle between two DMA request. 0: at least 1T 1: at least 2T 2: at least 4T 3: at least 8T 4: at least 16T 5: at least 32T 6: at least 64T 7: at least 128T
11:10		DMAW_BURST	**DMA Write Burst Size Configuraiton** This register configures the maximum burst size per DMA request to DRAM bus.

Bit(s)	Mnemonic	Name	Description
			0: 1024 bytes 1: 512 bytes 2: 256 bytes 3: 128 bytes
9		DMAW_BUFFERABLE	**DMA Write Bufferable Configuration** When this bit is set, the DMA write operation is bufferable on target DMA bus
8		DMAW_FAKE	**DMA Write Fake Configuration** When this bit is set, no data will be moved but DMA returns ack immediately.
6:4		DMAR_LIMITER	**DMA Read Limiter Configuration** This register configures the number of DRAM cycle between two DMA request. 0: at least 1T 1: at least 2T 2: at least 4T 3: at least 8T 4: at least 16T 5: at least 32T 6: at least 64T 7: at least 128T
3:2		DMAR_BURST	**DMA Read Burst Size Configuraiton** This register configures the maximum burst size per DMA request to DRAM bus. 0: 1024 bytes 1: 512 bytes 2: 256 bytes 3: 128 bytes
0		DMAR_FAKE	**DMA Read Fake Configuration** When this bit is set, no data will be moved but DMA returns ack immediately.

1A240954 ASYNC_HS_EOF Asynchronous High Speed EOF Start Offset 000100FA

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Nam													ASYNC_HS			

e													_EOF	_BAN	K	
Type													RW			
Reset													0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								ASYNC_HS_EOF_OFFSET								
Type								RW								
Reset								0	1	1	1	1	1	0	1	0

Bit(s)	Mnemonic	Name	Description
19:16		ASYNC_HS_EOF_BANK	**Asynchronous High Speed EOF Bank** One micro frame, 125 us, is divided into 15 banks which is 500 cycles of 60 MHz for each. This register set the bank that Asynchronous HSEOF asserts. The valid value can be 0 to 14.
8:0		ASYNC_HS_EOF_OFFSET	**Asynchronous High Speed EOF Offset** This register set the offset within a bank that Asynchronous HSEOF asserts. The valid value is 0~499.

1A240958 AXI_WR_DMA_CFG AXI WR DMA configuration register. 00401810

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									axi_wr_outstand_num				axi_wr_coherence	axi_wr_iommu	axi_wr_cacheable	axi_wr_ultra_en
Type									RW				RW	RW	RW	RW
Reset									0	1	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	axi_wr_ultra_num								axi_wr_pre_ultra_num							
Type	RW								RW							
Reset	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23:20		axi_wr_outstand_num	**The max outstand request for AXI DMA** valid value : 1 ~ 4

Bit(s)	Mnemonic	Name	Description
			The register are only for AXI DMA.
19		axi_wr_coheren ce	**AXI coherence capability.** 0 : disalbe 1 : enable
18		axi_wr_iommu	**AXI iommu capability.** 0 : disalbe 1 : enable
17		axi_wr_cacheab le	**AXI cache capability.** 0 : disalbe 1 : enable
16		axi_wr_ultra_e n	**AXI WR channel ultra capability which is used to indicate async fifo in AXI DMA is almost full.** 0 : disalbe 1 : enable
15:8		axi_wr_ultra_n um	**The high threshold to assert ultra signal. The valid can't be smaller than pre_ultra_num. When buffer data depth is more than the threshold, AXI DMA would assert ultra.** valid : 0 ~ 15
7:0		axi_wr_pre_ult ra_num	**The low threshold to assert pre_ultra signal. The valid can't be larger than ultra_num. When buffer data depth is more than the threshold, AXI DMA would assert pre_ultra.** valid : 0 ~ 15

1A24095C AXI_RD_DMA_CFG AXI RD DMA configuration register. 00401810

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name									axi_rd_outstand_n um				axi_rd_coherence	axi_rd_iommu	axi_rd_cacheable	axi_rd_ultra_en
Type									RW				RW	RW	RW	RW
Rese									0	1	0	0	0	0	0	0

t																	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	axi_rd_ultra_num								axi_rd_pre_ultra_num							
Type	RW								RW							
Reset	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0

Bit(s)	Mnemonic	Name	Description
23:20		axi_rd_outstand_num	**The max outstand request for AXI DMA** valid value : 1 ~ 4 The register are only for AXI DMA.
19		axi_rd_coheren ce	**AXI coherence capability.** 0 : disalbe 1 : enable
18		axi_rd_iommu	**AXI iommu capability.** 0 : disalbe 1 : enable
17		axi_rd_cacheab le	**AXI cache capability.** 0 : disalbe 1 : enable
16		axi_rd_ultra_e n	**AXI RD channel ultra capability which is used to indicate async fifo in AXI DMA is almost empty.** 0 : disalbe 1 : enable
15:8		axi_rd_ultra_n um	**The high threshold to assert ultra signal. The valid can't be smaller than pre_ultra_num. When available buffer space is more than the threshold, AXI DMA would assert ultra.** valid : 0 ~ 15
7:0		axi_rd_pre_ult ra_num	**The low threshold to assert pre_ultra signal. The valid can't be larger than ultra_num. When available buffer space is more than the threshold, AXI DMA would assert pre_ultra.** valid : 0 ~ 15

1A240960 HSCH_CFG1 Host Scheduler Configuration Register 1 003F1C20

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name											SCH3_RX_FIFO_DEPTH		SCH3_TX_FIFO_DEPTH		SCH2_FIFO_DEPTH	
Type											RW		RW		RW	
Reset											1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			PORT_IDLE_WITH_INT	SCH3_INT_PING_TD_CHK	SCH2_INT_PING_TD_CHK	SCH2_BULK_PING_TD_CHK	SCH_SPLIT_ISO_IN_RETRY_OPT		SCH_IN_ACK_RTY_EN	SCH_ASYNC_NEXT_FRAME	OUT_NUMP_REF_DYN	OUT_NUMP_REF	BURST_IN_OFF	BURST_TD_OFF	BURST_OUT_OFF	PORT_IDLE_WITH_ISO
Type			RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW	RW
Reset			0	1	1	1	0	0	0	0	1	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
21:20		SCH3_RX_FIFO_DEPTH	**Scheduler 3 RX Data FIFO Depth** This register control the available RX FIFO depth in Scheduler3 2'b11: 4KByte 2'b10: 3KByte 2'b01: 2KByte 2'b00: 1KByte Note: the default value matches the actual SRAM size. SW can only program this register with size smaller or equal to SRAM size.
19:18		SCH3_TX_FIFO_DEPTH	**Scheduler 3 TX Data FIFO Depth** This register control the available TX FIFO depth in Scheduler3 2'b11: 4KByte 2'b10: 3KByte 2'b01: 2KByte 2'b00: 1KByte Note: the default value matches the actual SRAM size. SW can only program this register with size smaller or equal to SRAM size.

Bit(s)	Mnemonic	Name	Description
17:16		SCH2_FIFO_DEPT H	**Scheduler 2 Data FIFO Depth** This register control the available FIFO depth in Scheduler2 2'b11: 4KByte 2'b10: 3KByte 2'b01: 2KByte 2'b00: 1KByte Note: the default value matches the actual SRAM size. SW can only program this register with size smaller or equal to SRAM size.
13		PORT_IDLE_WITH _INT	**Port Idle with INT Configuration** When this bit is set, the port_idle signal to transactor will gated with Interrupt endpoint active flags.
12		SCH3_INT_PING_ TD_CHK	**Scheduler 3 Interrupt Pseudo-PrePing TD Check** 1'b0: When Pseudo-PrePing is indicated by EP lookup, SCH3 directly wake link from U1/U2 to U0 whthout checking if there is TD to serve 1'b1: When Pseudo-PrePing is indicated by EP lookup, SCH3 only wake link from U1/U2 to U0 when the check result is with TD to serve
11		SCH2_INT_PING_ TD_CHK	**Scheduler 2 Interrupt Pseudo-PrePing TD Check** 1'b0: When Pseudo-PrePing is indicated by EP lookup, SCH2 directly wake link from U1/U2 to U0 whthout checking if there is TD to serve 1'b1: When Pseudo-PrePing is indicated by EP lookup, SCH2 only wake link from U1/U2 to U0 when the check result is with TD to serve
10		SCH2_BULK_PING _TD_CHK	**Scheduler 2 Bulk Pseudo-PrePing TD Check** 1'b0: When and OUT EP is with flow control, PING is sent without checking if there is OUT TD first. 1'b1: When and OUT EP is with flow control, PING is sent only when checking shows that there is OUT TD.
9:8		SCH_SPLIT_ISO_ IN_RETRY_OPT	**Scheduler 2 Split Iso In Retry Option** 2'b00: Whe transaction Error, retry 3 times 2'b01: Never Retry when transaction Error 2'b10:Infinitely retry until EOF when transaction Error 2'b11: Reserved

Bit(s)	Mnemonic	Name	Description
7		SCH_IN_ACK_RTY_EN	**Scheduler Send Ack with Retry set in 2nd Ack for IN Transfer** When this bit is set, the scheduler will request XACT3 to send Ack(NumP!=0, Retry=1) to resume IN transaction when this endpoint suffered DPP error by sending Aci(NumP=0, Retry=1) earlier
6		SCH_ASYNC_NEXT_FRAME	**Schedule Asynchrous EP in next micro Frame** When this bit is set, the scheduler will power down if no more asynchronous EP is active in current micro frame. The actived asynchronous EP might be delay serviced in the next micro frame.
5		OUT_NUMP_REF_DYN	**OUT Nump Reference Dynamically Configuration** When this bit is set, SSUSB OUT scheduler will reference the NUMP value in every ACK and not exceed device's capability dynamically.
4		OUT_NUMP_REF	**OUT Nump Reference Configuration** When this bit is set, SSUSB OUT scheduler will reference the NUMP value of last ERDY to determine the number of packets in this opportunity. It only applies to asynchronous endpoints.
3		BURST_IN_OFF	**Burst IN Off Configuration** When this bit is set, SSUSB IN scheduler will disable burst capability and launch IN ACK with NUMP = 1 only.
2		BURST_TD_OFF	**Burst TD Off Configuration** When this bit is set, SSUSB OUT scheduler will disable burst TD capability of bulk endpoint. It will end this endpoint service when current TD is finished no matter next endpoint is available.
1		BURST_OUT_OFF	**Burst OUT Off Configuration** When this bit is set, SSUSB OUT scheduler will disable burst capability and launch next OUT until ACK is returned.
0		PORT_IDLE_WITH_ISO	**Port Idle with ISO Configuration** When this bit is set, the port_idle signal to transactor

Bit(s)	Mnemonic	Name	Description
			will gated with isochronous endpoint active flags.

1A240964 CMD_CFG Command Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																PARAM_ERR_CHK_DIS
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0		PARAM_ERR_CHK_DIS	**Disable Parameter Error Check** When this bit is set, the parameter error check in command ring is disabled. Whe this be is not set, the parameter error check in command ring is enabled.

1A240968 EP_CFG Endpoint Status Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																HUB_TTT_EN

Type															RW
Reset															0

Bit(s)	Mnemonic	Name	Description
0		HUB_TTT_EN	**Enable TT Thing Time of HUB** When this bit is set, the EP will read related TT Hub's TT Think Time when serving a Lookup request and forward it to SCH2 and then XACT2. Whe this be is not set, the EP does not read related TT Hub's TT Think Time for Lookup request and the XACT2 simply ignores the TT Think Time provided.

1A24096C EVT_CFG Event Configuration 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																INTR_BEI_EN
Type																RW
Reset																1

Bit(s)	Mnemonic	Name	Description
0		INTR_BEI_EN	**Enable BEI Support for Interrupt** When this bit is set, the BEI (Block Event Interrupt) feature is supported Whe this be is not set, the BEI (Block Event Interrupt) feature is not supported

1A240970 TRBQ_CFG TRBQ Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Na																

me																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																TRB_ERR_CHK_DIS
Type																RW
Reset																0

Bit(s)	Mnemonic	Name	Description
0		TRB_ERR_CHK_DI S	**Disable TRB Error Check** When this bit is set, the TRB error check in command ring is disabled. Whe this be is not set, the TRB error check in command ring is enabled.

1A240974 U3PORT_CFG USB3 Port Configuration 00000001

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name																U3PORT_WRP_BY_HCRST_EN
Type																RW
Reset																1

Bit(s)	Mnemonic	Name	Description

Bit(s)	Mnemonic	Name	Description
0		U3PORT_WRP_BY_HCRST_EN	**Enable USB3 Port Warm Reset by HCRST** When this bit is set, the USB3 Port will send Warm Reset Whe this be is not set, the the USB3 Port will not send Warm Reset

1A240978 U2PORT_CFG USB2 Port Configuration 00000004

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name									LPM_L1_EXIT_TIMER							
Type									RW							
Reset									0	0	0	0	0	1	0	0

Bit(s)	Mnemonic	Name	Description
7:0		LPM_L1_EXIT_TIMER	**LPM L1 Exit Timer** This register indicate the no. of uFrame to stay in L1 if there is no EP are masked by NYET.

1A24097C HSCH_CFG2 Host Scheduler Configuration Register 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				SCH_INT_NAK_ACTI	SCH_INT_NAK_ACTI	SCH_INT_NAK_ACTI	SCH_INT_NAK_ACTI	SCH_INT_NAK_ACTI				SCH_BULK_NYET_AC	SCH_BULK_NYET_AC	SCH_BULK_NYET_AC	SCH_BULK_NYET_AC	SCH_BULK_NYET_AC

				VE_MASK_4P	VE_MASK_3P	VE_MASK_2P	VE_MASK_1P	VE_MASK				TIVE_MASK_4P	TIVE_MASK_3P	TIVE_MASK_2P	TIVE_MASK_1P	TIVE_MASK
Type				RW	RW	RW	RW	RW				RW	RW	RW	RW	RW
Reset				0	0	0	0	0				0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
12		SCH_INT_NAK_ACTIVE_MASK_4P	**Scheduler 2 Port 4 Active Mask when Interrupt is NAKed** 1'b0: Active Mask is not set when Interrupt is NAKed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Interrupt is NAKed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
11		SCH_INT_NAK_ACTIVE_MASK_3P	**Scheduler 2 Port 3 Active Mask when Interrupt is NAKed** 1'b0: Active Mask is not set when Interrupt is NAKed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Interrupt is NAKed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
10		SCH_INT_NAK_ACTIVE_MASK_2P	**Scheduler 2 Port 2 Active Mask when Interrupt is NAKed** 1'b0: Active Mask is not set when Interrupt is NAKed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Interrupt is NAKed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
9		SCH_INT_NAK_ACTIVE_MASK_1P	**Scheduler 2 Port 1 Active Mask when Interrupt is NAKed** 1'b0: Active Mask is not set when Interrupt is NAKed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Interrupt is NAKed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
8		SCH_INT_NAK_ACTIVE_MASK	**Scheduler 2 Port 0 Active Mask when Interrupt is NAKed** 1'b0: Active Mask is not set when Interrupt is NAKed. xHC can actively exilt USB2 LPM

Bit(s)	Mnemonic	Name	Description
			1'b1: Active Mask is set when Interrupt is NAKed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
4		SCH_BULK_NYET_ACTIVE_MASK_4P	**Scheduler 2 Port 4 Active Mask when Bulk OUT is NYETed** 1'b0: Active Mask is not set when Bulk OUT is NYETed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Bulk OUT is NYETed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
3		SCH_BULK_NYET_ACTIVE_MASK_3P	**Scheduler 2 Port 3 Active Mask when Bulk OUT is NYETed** 1'b0: Active Mask is not set when Bulk OUT is NYETed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Bulk OUT is NYETed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
2		SCH_BULK_NYET_ACTIVE_MASK_2P	**Scheduler 2 Port 2 Active Mask when Bulk OUT is NYETed** 1'b0: Active Mask is not set when Bulk OUT is NYETed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Bulk OUT is NYETed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
1		SCH_BULK_NYET_ACTIVE_MASK_1P	**Scheduler 2 Port 1 Active Mask when Bulk OUT is NYETed** 1'b0: Active Mask is not set when Bulk OUT is NYETed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Bulk OUT is NYETed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup
0		SCH_BULK_NYET_ACTIVE_MASK	**Scheduler 2 Port 0 Active Mask when Bulk OUT is NYETed** 1'b0: Active Mask is not set when Bulk OUT is NYETed. xHC can actively exilt USB2 LPM 1'b1: Active Mask is set when Bulk OUT is NYETed. xHC cannot actively exilt USB2 LPM. Wait remove LPM wakeup

1A240980 SW_ERDY Software ERDY 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SW_ERDY_CMD			SW_ERDY_DCI					SW_ERDY_SLOT_ID							
Type	A0			RW					RW							
Reset	0			0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15		SW_ERDY_CMD	**Software ERDY CMD** When set to 1'b1, software ERDY with target Slot ID and DCI are requested When xHC complete processing of the software ERDY, this bit is cleared by HW. Note that this request only take effect when the endpoint state is Running and is flow controlled state, otherwise the request will be ignored.
12:8		SW_ERDY_DCI	**Software ERDY DCI** When SOFT_ERDY_CMD is set to 1'b1, this represents the DCI of software ERDY. For EP0 of a device, DCI shall be 1
7:0		SW_ERDY_SLOT_ID	**Software ERDY Slot ID** When SOFT_ERDY_CMD is set to 1'b1, this represents the Slot ID of software ERDY. Software shall guarantee the Slot ID of the device is operating at Super Speed, otherwise undefined result may happen. The software must al guarantees the Slot ID cannot be 0's

1A2409A0 SLOT_EP_STS0 Slot and EP Resource Status0 00000F40

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Na																

me																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AVAIL_SLOT_NUM								AVAIL_EP_NUM							
Type	RO								RO							
Reset	0	0	0	0	1	1	1	1	0	1	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:8		AVAIL_SLOT_NUM	**Available Slot Number** This register indicates the remaining Slot number to be allocated for Enable Slot command
7:0		AVAIL_EP_NUM	**Available EP Number** This register indicates the remaining EP number to be allocated for Address Device and Configure Endpoint commands

1A2409A4 SLOT_EP_STS1 Slot and EP Resource Status1 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AVAIL_EP_BITMAP_LO															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AVAIL_EP_BITMAP_LO															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		AVAIL_EP_BITMAP_LO	**Available EP Bitmap Lo** This register represned whether EP index 31~0 are available for SW to add endpoint. A bit of value equal to 1'b1, means that this EP index is still available

1A2409A8 SLOT_EP_STS2 Slot and EP Resource Status2 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	AVAIL_EP_BITMAP_HI															

Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	AVAIL_EP_BITMAP_HI															
Type	RO															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		AVAIL_EP_BITMAP_HI	**Available EP Bitmap Hi** This register represned whether EP index 63~32 are available for SW to add endpoint. A bit of value equal to 1'b1, means that this EP index is still available

1A2409B0 RST_CTRL0 Host reset control Register 2 0000000F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													HCRST_DRAM_EN	HCRST_SYS60_EN	HCRST_SYS125_EN	HCRST_XHCI_EN
Type													RW	RW	RW	RW
Reset													1	1	1	1

Bit(s)	Mnemonic	Name	Description
3		HCRST_DRAM_EN	**dram_rst_b by HCRST Enable** When this bit is 1'b1, the HCRST results in dram_rst_b
2		HCRST_SYS60_EN	**sys60_rst_b by HCRST Enable** When this bit is 1'b1, the HCRST results in sys60_rst_b
1		HCRST_SYS125_EN	**sys125_rst_b by HCRST Enable** When this bit is 1'b1, the HCRST results in sys125_rst_b

Bit(s)	Mnemonic	Name	Description
0		HCRST_XHCI_EN	**xhci_rst_b by HCRST Enable** When this bit is 1'b1, the HCRST results in xhci_rst_b

1A2409B4 RST_CTRL1 Host reset control Register 3 03FF000F

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name							HCRST_U2_PHYD_EN_4P	HCRST_U2_MAC_EN_4P	HCRST_U2_PHYD_EN_3P	HCRST_U2_MAC_EN_3P	HCRST_U2_PHYD_EN_2P	HCRST_U2_MAC_EN_2P	HCRST_U2_PHYD_EN_1P	HCRST_U2_MAC_EN_1P	HCRST_U2_PHYD_EN_0P	HCRST_U2_MAC_EN_0P
Type							RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset							1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name													HCRST_U3_PHYD_EN_1P	HCRST_U3_MAC_EN_1P	HCRST_U3_PHYD_EN_0P	HCRST_U3_MAC_EN_0P
Type													RW	RW	RW	RW
Reset													1	1	1	1

Bit(s)	Mnemonic	Name	Description
25		HCRST_U2_PHYD_EN_4P	**Enable HCRST as software reset for USB2 PHYD Port4** 1'b0: HCRST does not imply software reest for target USB2 PHYD 1'b1: HCRST implies software reset for target USB2 PHYD
24		HCRST_U2_MAC_EN_4P	**Enable HCRST as software reset for USB2 MAC Port4** 1'b0: HCRST does not imply software reest for target USB2 MAC 1'b1: HCRST implies software reset for target USB2 MAC

Bit(s)	Mnemonic	Name	Description
23		HCRST_U2_PHYD_EN_3P	**Enable HCRST as software reset for USB2 PHYD Port3** 1'b0: HCRST does not imply software reest for target USB2 PHYD 1'b1: HCRST implies software reset for target USB2 PHYD
22		HCRST_U2_MAC_EN_3P	**Enable HCRST as software reset for USB2 MAC Port3** 1'b0: HCRST does not imply software reest for target USB2 MAC 1'b1: HCRST implies software reset for target USB2 MAC
21		HCRST_U2_PHYD_EN_2P	**Enable HCRST as software reset for USB2 PHYD Port2** 1'b0: HCRST does not imply software reest for target USB2 PHYD 1'b1: HCRST implies software reset for target USB2 PHYD
20		HCRST_U2_MAC_EN_2P	**Enable HCRST as software reset for USB2 MAC Port2** 1'b0: HCRST does not imply software reest for target USB2 MAC 1'b1: HCRST implies software reset for target USB2 MAC
19		HCRST_U2_PHYD_EN_1P	**Enable HCRST as software reset for USB2 PHYD Port1** 1'b0: HCRST does not imply software reest for target USB2 PHYD 1'b1: HCRST implies software reset for target USB2 PHYD
18		HCRST_U2_MAC_EN_1P	**Enable HCRST as software reset for USB2 MAC Port1** 1'b0: HCRST does not imply software reest for target USB2 MAC 1'b1: HCRST implies software reset for target USB2 MAC
17		HCRST_U2_PHYD_EN_0P	**Enable HCRST as software reset for USB2 PHYD Port0**

Bit(s)	Mnemonic	Name	Description
			1'b0: HCRST does not imply software reest for target USB2 PHYD 1'b1: HCRST implies software reset for target USB2 PHYD
16		HCRST_U2_MAC_EN_0P	**Enable HCRST as software reset for USB2 MAC Port0** 1'b0: HCRST does not imply software reest for target USB2 MAC 1'b1: HCRST implies software reset for target USB2 MAC
3		HCRST_U3_PHYD_EN_1P	**Enable HCRST as software reset for USB3 PHYD Port1** 1'b0: HCRST does not imply software reest for target USB3 PHYD 1'b1: HCRST implies software reset for target USB3 PHYD
2		HCRST_U3_MAC_EN_1P	**Enable HCRST as software reset for USB3 MAC Port1** 1'b0: HCRST does not imply software reest for target USB3 MAC 1'b1: HCRST implies software reset for target USB3 MAC
1		HCRST_U3_PHYD_EN_0P	**Enable HCRST as software reset for USB3 PHYD Port0** 1'b0: HCRST does not imply software reest for target USB3 PHYD 1'b1: HCRST implies software reset for target USB3 PHYD
0		HCRST_U3_MAC_EN_0P	**Enable HCRST as software reset for USB3 MAC Port0** 1'b0: HCRST does not imply software reest for target USB3 MAC 1'b1: HCRST implies software reset for target USB3 MAC

1A2409F0 SPARE0 Spare Register 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	SPARE0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPARE0															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:0		SPARE0	**Spare Register 0** This register is with default value 0's and is prepared for ECO purpose

1A2409F4 SPARE1 Spare Register 1 FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SPARE1															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SPARE1															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Mnemonic	Name	Description
31:0		SPARE1	**Spare Register 1** This register is with default value 1's and is prepared for ECO purpose

20.3.2 ssusb2_xhci_u2_port_csr

Module name: ssusb2_xhci_u2_port_csr Base address: (+1a240440h)

Address	Name	Width	Register Function
1a240440	USB2_PORT_SC	32	USB2_PORT Port Status and Control
1a240444	USB2_PORT_PMSC	32	USB2_PORT PM Status and Control
1a240448	USB2_PORT_LI	32	USB2_PORT Link Info
1a24044c	USB2_PORT_HLPMC	32	USB2_PORT Hardware LPM Control Register

1A240440 USB2_PORT_SC USB2_PORT Port Status and Control 000002A0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB2_PORT_WPR	USB2_PORT_DR			USB2_PORT_WOE	USB2_PORT_WDE	USB2_PORT_WCE		USB2_PORT_CEC	USB2_PORT_PLC	USB2_PORT_PRC	USB2_PORT_OCC	USB2_PORT_WRC	USB2_PORT_PEC	USB2_PORT_CSC	USB2_PORT_LWS
Type	RO	RO			RW	RW	RW		RO	W1C	W1C	W1C	W1C	W1C	W1C	WO
Reset	0	0			0	0	0		0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	USB2_PORT_PIC		USB2_PORT_Port_Speed				USB2_PORT_PP	USB2_PORT_PLS				USB2_PORT_PR	USB2_PORT_OCA		USB2_PORT_PED	USB2_PORT_CCS
Type	RW		RO				RW	RW				OTHER	RO		W1C	RO
Reset	0	0	0	0	0	0	1	0	1	0	1	0	0		0	0

Bit(s)	Mnemonic	Name	Description
31		USB2_PORT_WPR	**Warm Port Reset (WPR).** Default = '0'. When software writes a '1' to this bit, the Warm Reset sequence as defined in the USB3 Specification is initiated and the PR flag is set to '1'. Once initiated, the PR, PRC, and WRC flags shall reflect the progress of the Warm Reset sequence. This flag shall always return '0' when read. Refer to section 4.19.5.1. This flag only applies to USB3 protocol ports. For USB2 protocol ports it shall be RsvdZ
30		USB2_PORT_DR	**Device Removablej (DR).** This flag indicates if this port has a removable device attached. '1' = Device is non-removable. '0' = Device is removable.
27		USB2_PORT_WOE	**Wake on Over-current Enable (WOE).** Default = '0'. Writing this bit to a '1' enables the port to be sensitive to over-current conditions as system wake-up eventsi. Refer to section 4.15 for operational model.
26		USB2_PORT_WDE	**Wake on Disconnect Enable (WDE).** Default = '0'. Writing this bit to a '1' enables the port to be sensitive to device disconnects as system wake-up eventsi. Refer to section 4.15 for operational model.

Bit(s)	Mnemonic	Name	Description
25		USB2_PORT_WCE	**Wake on Connect Enable (WCE).Default = '0'. Writing this bit to a '1' enables the port to be sensitive to device connects as system wake-up eventsi. Refer to section 4.15 for operational model.**
23		USB2_PORT_CEC	**Port Config Error Change (CEC).** Default = '0'. This flag indicates that the port failed to configure its link partner. 0 = No change. 1 = Port Config Error detected. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage. Note: This flag is valid only for USB3 protocol ports. For USB2 protocol ports this bit shall be RsvdZ.
22		USB2_PORT_PLC	**Port Link State Change (PLC).** Default = '0'. This flag is set to '1' due to the following PLS transitions: Transition Condition U3 -> Resume Wakeup signaling from a device Resume -> Recovery -> U0 Device Resume complete (USB3 protocol ports only) Resume -> U0 Device Resume complete (USB2 protocol ports only) U3 -> Recovery -> U0 Software Resume complete (USB3 protocol ports only) U3 -> U0 Software Resume complete (USB2 protocol ports only) U2 -> U0 L1 Resume complete (USB2 protocol ports only)h U0 -> U0 L1 Entry Reject (USB2 protocol ports only)h Any state -> Inactive Error (USB3 protocol ports only). Note: PLC is asserted only if there is an SS.Inactive.Disconnect.Detect to SS.Inactive.Quiet transition in the LTSSM. Note that this flag shall not be set if the PLS transition was due to software setting PP to '0'. Refer to section 4.23.5 for more information. '0' = No change. '1' = Link Status Changed. Software shall clear this bit by writing a '1' to it. Refer to "PLC Condition:" references in section 4.19.1 for the specific port state transitions that set this flag. Refer to section 4.19.2 for more information on change bit usage.

Bit(s)	Mnemonic	Name	Description
21		USB2_PORT_PRC	**Port Reset Change (PRC).** Default = '0'. This flag is set to '1' due a '1' to '0' transition of Port Reset (PR). e.g. when any reset processing (Warm or Hot) on this port is complete. Note that this flag shall not be set to '1' if the reset processing was forced to terminate due to software clearing PP or PED to '0'. '0' = No change. '1' = Reset complete. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.5. Refer to section 4.19.2 for more information on change bit usage.
20		USB2_PORT_OCC	**Over-current Change (OCC).** Default = '0'. This bit shall be set to a '1' when there is a '0' to '1' or '1' to '0' transition of Over-current Active (OCA). Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage.
19		USB2_PORT_WRC	**Warm Port Reset Change (WRC).** Default = '0'. This bit is set when Warm Reset processing on this port completes. '0' = No change. '1' = Warm Reset complete. Note that this flag shall not be set to '1' if the Warm Reset processing was forced to terminate due to software clearing PP or PED to '0'. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.5.1. Refer to section 4.19.2 for more information on change bit usage. This bit only applies to USB3 protocol ports. For USB2 protocol ports it shall be RsvdZ.
18		USB2_PORT_PEC	**Port Enabled/Disabled Change (PEC).** Default = '0'. '1' = change in PED. '0' = No change. Note that this flag shall not be set if the PED transition was due to software setting PP to '0'. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage. For a USB2 protocol port, this bit shall be set to '1' only when the port is disabled due to the appropriate conditions existing at the EOF2 point (refer to section 11.8.1 of the USB2 Specification for the definition of a Port Error). For a USB3 protocol port, this bit shall never be set to '1'.
17		USB2_PORT_CSC	**Connect Status Change (CSC).** Default = '0'. '1' = Change in CCS. '0' = No change. This

Bit(s)	Mnemonic	Name	Description
			flag indicates a change has occurred in the port's Current Connect Status (CCS). Note that this flag shall not be set if the CCS transition was due to software setting PP to '0'. The xHC sets this bit to '1' for all changes to the port device connect status, even if system software has not cleared an existing Connect Status Change. For example, the insertion status changes twice before system software has cleared the changed condition, root hub hardware will be "setting" an already-set bit (i.e., the bit will remain '1'). Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage.
16		USB2_PORT_LWS	**Port Link State Write Strobe (LWS).** Default = '0'. When this bit is set to '1' on a write reference to this register, this flag enables writes to the PLS field. When '0', write data in PLS field is ignored. Reads to this bit return '0'.
15:14		USB2_PORT_PIC	**Port Indicator Control (PIC).** Default = 0. Writing to these bits has no effect if the Port Indicators (PIND) bit in the HCCPARAMS register is a '0'. If PIND bit is a '1', then the bit encodings are: Value Meaning 0 Port indicators are off 1 Amber 2 Green 3 Undefined Refer to the USB2 Specification for a description on how these bits are to be used. This field is '0' if PP is '0'.
13:10		USB2_PORT_Port_Speed	**Port Speed (Port Speed).** Default = '0'. This field identifies the speed of the attached USB Device. This field is only relevant if a device is attached (CCS = '1') in all other cases this field shall indicate Undefined Speed. Value Meaning 0 Undefined Speed 1 Full-speed device attached 2 Low-speed device attached 3 High-speed device attached 4 SuperSpeed device attached 5-15 Reserved Note: Values the 1, 2 and 3 are exclusive to USB2 protocol ports and the value 4 is exclusive to USB3 protocol ports

Bit(s)	Mnemonic	Name	Description
9		USB2_PORT_PP	**Port Power (PP).** Default = '1'. This flag reflects a port's logical, power control state. Because host controllers can implement different methods of port power switching, this flag may or may not represent whether (VBus) power is actually applied to the port. When PP equals a '0' the port is nonfunctional and shall not report attaches, detaches, or Port Link State (PLS) changes. However, the port shall report over-current conditions when PP = '0' if PPC = '0'. 0 = This port is in the Powered-off state. 1 = This port is not in the Powered-off state. If the Port Power Control (PPC) flag in the HCCPARAMS register is '1', then xHC has port power control switches and this bit represents the current setting of the switch ('0' = off, '1' = on). If the Port Power Control (PPC) flag in the HCCPARAMS register is '0', then xHC does not have port power control switches and each port is hard wired to power, and not affected by this bit. When an over-current condition is detected on a powered port, the xHC shall transition the PP bit in each affected port from a '1' to '0' (removing power from the port). Refer to section 4.19.4 for more information.
8:5		USB2_PORT_PLS	**Port Link State (PLS).** Default = RxDetect ('5'). This field is used to power manage the port and reflects its current link state. When the port is in the Enabled state, system software may set the link U state by writing this field. System software may also write this field to force a Disabled to Disconnected state transition of the port. Write Value Description 0 The link shall transition to a U0 state from any of the U states. 2b USB2 protocol ports only. The link should transition to the U2 State. 3 The link shall transition to a U3 state from any of the U states. This action selectively suspends the device connected to this port. While the Port Link State = U3, the hub does not propagate downstream-directed traffic to this port, but the hub will respond to resume signaling from the port. 5 USB3 protocol ports only. If the port is in the Disabled

Bit(s)	Mnemonic	Name	Description
			state (PLS = Disabled, PP = 1), then the link shall transition to a RxDetect state and the port shall transition to the Disconnected state, else ignored. 1b,4,6-14 Ignored. 15 USB2 protocol ports only. If the port is in the U3 state (PLS = U3), then the link shall remain in the U3 state and the port shall transition to the U3Exit substate, else ignored. Refer to section 4.15.2 for more information. Note: The Port Link State Write Strobe (LWS) shall also be set to '1' to write this field. For USB2 protocol ports: Writing a value of '2' to this field shall request LPM, asserting L1 signaling on the USB2 bus. Software may read this field to determine if the transition to the U2 state was successful. Writing a value of '0' shall deassert L1 signaling on the USB. Writing a value of '1' shall have no affect. The U1 state shall never be reported by a USB2 protocol port. Read Value Meaning 0 Link is in the U0 State 1 Link is in the U1 State 2 Link is in the U2 State 3 Link is in the U3 State (Device Suspended) 4 Link is in the Disabled Statec 5 Link is in the RxDetect Stated 6 Link is in the Inactive Statee 7 Link is in the Polling State 8 Link is in the Recovery State 9 Link is in the Hot Reset State 10 Link is in the Compliance Mode State 11 Link is in the Test Modef State 12:14 Reserved 15 Link is in the Resume Stateg This field is undefined if PP = '0'. Note: Transitions between different states are not reflected until the transition is complete. Refer to sections 4.15.2 and 4.23.5 for more information on the use of this field. Refer to the USB2 LPM ECR for more information on USB link power management operation. Refer to section 7.2 for supported USB protocols.
4		USB2_PORT_PR	**Port Reset (PR).** Default = '0'. '1' = Port Reset signaling is asserted. '0' = Port is not in Reset. When software writes a '1' to this bit (from a '0') the bus reset sequence is initiated; USB2 protocol ports shall execute the bus reset sequence as defined in the USB2 Spec. USB3 protocol ports shall execute the Hot Reset sequence as defined in the USB3 Spec. PR remains set until reset signaling is

Bit(s)	Mnemonic	Name	Description
			completed by the root hub. Note that software shall write a '1' to the this flag to transition a USB2 port from the Polling state to the Enabled state. Refer to sections 4.15.2.3 and 4.19.1.1. This flag is '0' if PP is '0'.
3		USB2_PORT_OCA	**Over-current Active (OCA).** Default = '0'. '1' = This port currently has an over-current condition. '0' = This port does not have an over-current condition. This bit shall automatically transition from a '1' to a '0' when the over-current condition is removed.
1		USB2_PORT_PED	**Port Enabled/Disabled (PED).** Default = '0'. '1' = Enabled. '0' = Disabled. Ports may only be enabled by the xHC. Software cannot enable a port by writing a '1' to this flag. A port may be disabled by software writing a '1' to this flag. This flag shall automatically be cleared to '0' by a disconnect event or other fault condition. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller or bus events. When the port is disabled (PED = '0') downstream propagation of data is blocked on this port, except for reset. For USB2 protocol ports: When the port is in the Disabled state, software shall reset the port (PR = '1') to transition PED to '1' and the port to the Enabled state. For USB3 protocol ports: When the port is in the Polling state (after detecting an attach), the port shall automatically transition to the Enabled state and set PED to '1' upon the completion of successful link training. When the port is in the Disabled state, software shall write a '5' (RxDetect) to the PLS field to transition the port to the Disconnected state. Refer to section 4.19.1.2. PED shall automatically be cleared to '0' when PR is set to '1', and set to '1' when PR transitions from '1' to '0' after a successful reset. Refer to Port Reset (PR) bit for more information on how the PED bit is managed. Note that when software writes this bit to a '1', it shall also write a '0' to the PR bita. This flag is '0' if PP is '0'.

Bit(s)	Mnemonic	Name	Description
0		USB2_PORT_CCS	**Current Connect Status (CCS).** Default = '0'. '1' = Device is present on port. '0' = No device is present. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change (CSC) bit to be set to '1'. Refer to section 4.19.4 for more details on the Connect Status Change (CSC) assertion conditions. This flag is '0' if PP is '0'.

1A240444 USB2_PORT_PMSC USB2_PORT PM Status and Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB2_PORT_Port_Test_Ctrl															USB2_PORT_HW_LPM_EN
Type	RW															RW
Reset	0	0	0	0												0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	USB2_PORT_L1_Device_Slot								USB2_PORT_BESL				USB2_PORT_RWE	USB2_PORT_L1S		
Type	RW								RW				RW	RO		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
31:28		USB2_PORT_Port_Test_Ctrl	**Port Test Control.** Default = '0'. When this field is '0', the port is NOT operating in a test mode. A non-zero value indicates that it is operating in test mode and the specific test mode is indicated by the specific value. A non-zero Port Test Control value is only valid to a port that is in the Powered-Off state (PLS = Disabled). If the port is not in this state, the xHC shall respond with the Port Test Control field set to Port Test Control Error. Refer to section 4.19.6 for the operational model for using these test

Bit(s)	Mnemonic	Name	Description
			modes. The encoding of the Test Mode bits for a USB2 protocol port are: Value Test Mode 0 Test mode not enabled 1 Test J_STATE 2 Test K_STATE 3 Test SE0_NAK 4 Test Packet 5 Test FORCE_ENABLE 6-14 Reserved. 15 Port Test Control Error. Refer to the sections 7.1.20 and 11.24.2.13 of the USB2 spec for more information on Test Modes.
16		USB2_PORT_HW_LPM_EN	**Hardware LPM Enable** Default = '0'. If this bit is set to '1', then hardware controlled LPM shall be enabled for this port. Refer to section 4.23.5.1.1.1. If the USB2 Hardware LMP Capability is not supported (HLC = '0') this field shall be RsvdZ.
15:8		USB2_PORT_L1_Device_Slot	**L1 Device Slot.** Default = '0'. System software sets this field to indicate the ID of the Device Slot associated with the device directly attached to the Root Hub port. A value of '0' indicates no device is present. The xHC uses this field to lookup information necessary to generate the LMP Token packet. If the USB2 L1 Capability is not supported (L1C = '0') this field shall be RsvdZ.
7:4		USB2_PORT_BESL	**Best Effort Service Latency (BESL).** Default = '0'. System software sets this field to indicate to the recipient device how long the xHC will drive resume if it (the xHC) initiates an exit from L1. The BESL value encoding is defined in Table 13. Note that the BESL field is used by both software and hardware controlled LPM. Refer to section 4.23.5.1.1 for more information on BESL use. Refer to section 5.2.5 for information on how DBESL may be used to establish an initial value for

Bit(s)	Mnemonic	Name	Description
			BESL.
3		USB2_PORT_RWE	**Remote Wake Enable (RWE).** Default = '0'. The host system sets this flag to enable or disable the device for remote wake from L1. The value of this flag will temporarily (while in L1) override the current setting of the Remote Wake feature set by the standard Set/ClearFeature() commands defined in Universal Serial Bus Specification, revision 2.0, Chapter 9.
2:0		USB2_PORT_L1S	**L1 Status (L1S).** Default = 0. This field is used by software to determine whether an L1- based suspend request (LMP transaction) was successful, specifically: Value Meaning 0 Invalid - This field shall be ignored by software. 1 Success - Port successfully transitioned to L1 (ACK) 2 Not Yet - Device is unable to enter L1 at this time (NYET) 3 Not Supported - Device does not support L1 transitions (STALL) 4 Timeout/Error - Device failed to respond to the LPM Transaction or an error occurred 5-7 Reserved The value of this field is only valid when the port resides in the L0 or L1 state (PLS = '0' or '2'). Refer to section 4.23.5.1.1 for more information. If the USB2 L1 Capability is not supported (L1C = '0') this field shall be RsvdZ.

1A240448 USB2_PORT_LI USB2_PORT Link Info 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**USB2_PORT_Link_Error_Count**															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		USB2_PORT_Link_Error_Count	**Link Error Count.** Default = '0'. This field returns the number of link errors detected by the port. This value shall be reset to '0' by the assertion of a Chip Hardware Reset, HCRST, when PR transitions from '1' to '0', or when CCS = transitions from '0' to '1'.

1A24044C USB2_PORT_HLPMC USB2_PORT Hardware LPM Control Register 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name			USB2_PORT_BESLD				USB2_PORT_L1_TIMEOUT								USB2_PORT_HIRDM	
Type			RW				RW								RW	
Reset			0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
13:10		USB2_PORT_BESLD	**Best Effort Service Latency Deep (BESLD) - RWS. Default = '0'. System software sets this** field to indicate to the recipient device how long the xHC will drive resume on an exit from U2. Refer to section 4.23.5.1.1.1 for more information on BESLD use. The BESLD value encoding is defined in Table 13. Refer to section 5.2.6 for information on how DBESLD may be used to establish an initial value for BESLD.
9:2		USB2_PORT_L1_TIMEOUT	**L1 Timeout - RWS. Default = 00h. Timeout value for the L1 inactivity timer (LPM Timer). This**

Bit(s)	Mnemonic	Name	Description
			field shall be set to 00h by the assertion of PR to '1'. Refer to section 4.23.5.1.1.1 for more information on L1 Timeout operation. The following are permissible values: Value Description 00h 128 us. (default) 01h 256 us. 02h 512 us. 03h 768 us. ... FFh 65,280 us.
1:0		USB2_PORT_HIRD M	**Host Initiated Resume Duration Mode (HIRDM) - RWS. Default = 0h. Indicates which HIRD** value should be used. The following are permissible values: 0 Initiate L1 using BESL only on timeout. (default) 1 Initiate L1 using BESLD on timeout. If rejected by device, initiate L1 using BESL. 3-2 Reserved.

20.3.3 ssusb2_xhci_u3_port_csr

Module name: ssusb2_xhci_u3_port_csr Base address: (+1a240420h)

Address	Name	Width	Register Function
1a240420	USB3_PORT_SC	32	USB3_PORT Port Status and Control
1a240424	USB3_PORT_PMSC	32	USB3_PORT PM Status and Control
1a240428	USB3_PORT_LI	32	USB3_PORT Link Info

1A240420 USB3_PORT_SC USB3_PORT Port Status and Control 000002A0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	USB3_PORT_WPR	USB3_PORT_DR			USB3_PORT_WOE	USB3_PORT_WDE	USB3_PORT_WCE		USB3_PORT_CEC	USB3_PORT_PLC	USB3_PORT_PRC	USB3_PORT_OCC	USB3_PORT_WRC	USB3_PORT_PEC	USB3_PORT_CSC	USB3_PORT_LWS
Type	WO	RO			RW	RW	RW		W1C	W1C	W1C	W1C	W1C	RO	W1C	WO

Reset	0	0			0	0	0		0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	**USB3_PORT_PIC**		**USB3_PORT_Port_Speed**				**USB3_PORT_PP**	**USB3_PORT_PLS**				**USB3_PORT_PR**	**USB3_PORT_OCA**		**USB3_PORT_PED**	**USB3_PORT_CCS**
Type	RW		RO				RW	RW				OTHER	RO		W1C	RO
Reset	0	0	0	0	0	0	1	0	1	0	1	0	0		0	0

Bit(s)	Mnemonic	Name	Description
31		USB3_PORT_WPR	**Warm Port Reset (WPR).** Default = '0'. When software writes a '1' to this bit, the Warm Reset sequence as defined in the USB3 Specification is initiated and the PR flag is set to '1'. Once initiated, the PR, PRC, and WRC flags shall reflect the progress of the Warm Reset sequence. This flag shall always return '0' when read. Refer to section 4.19.5.1. This flag only applies to USB3 protocol ports. For USB2 protocol ports it shall be RsvdZ
30		USB3_PORT_DR	**Device Removablej (DR).** This flag indicates if this port has a removable device attached. '1' = Device is non-removable. '0' = Device is removable.
27		USB3_PORT_WOE	**Wake on Over-current Enable (WOE).** Default = '0'. Writing this bit to a '1' enables the port to be sensitive to over-current conditions as system wake-up eventsi. Refer to section 4.15 for operational model.
26		USB3_PORT_WDE	**Wake on Disconnect Enable (WDE).** Default = '0'. Writing this bit to a '1' enables the port to be sensitive to device disconnects as system wake-up eventsi. Refer to section 4.15 for operational model.
25		USB3_PORT_WCE	**Wake on Connect Enable (WCE).** Default = '0'. Writing this bit to a '1' enables the port to be sensitive to device connects as system wake-up eventsi. Refer to section 4.15 for operational model.
23		USB3_PORT_CEC	**Port Config Error Change (CEC).** Default = '0'. This flag indicates that the port failed to

Bit(s)	Mnemonic	Name	Description
			configure its link partner. 0 = No change. 1 = Port Config Error detected. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage. Note: This flag is valid only for USB3 protocol ports. For USB2 protocol ports this bit shall be RsvdZ.
22		USB3_PORT_PLC	**Port Link State Change (PLC).** Default = '0'. This flag is set to '1' due to the following PLS transitions: Transition Condition U3 -> Resume Wakeup signaling from a device Resume -> Recovery -> U0 Device Resume complete (USB3 protocol ports only) Resume -> U0 Device Resume complete (USB2 protocol ports only) U3 -> Recovery -> U0 Software Resume complete (USB3 protocol ports only) U3 -> U0 Software Resume complete (USB2 protocol ports only) U2 -> U0 L1 Resume complete (USB2 protocol ports only)h U0 -> U0 L1 Entry Reject (USB2 protocol ports only)h Any state -> Inactive Error (USB3 protocol ports only). Note: PLC is asserted only if there is an SS.Inactive.Disconnect.Detect to SS.Inactive.Quiet transition in the LTSSM. Note that this flag shall not be set if the PLS transition was due to software setting PP to '0'. Refer to section 4.23.5 for more information. '0' = No change. '1' = Link Status Changed. Software shall clear this bit by writing a '1' to it. Refer to "PLC Condition:" references in section 4.19.1 for the specific port state transitions that set this flag. Refer to section 4.19.2 for more information on change bit usage.
21		USB3_PORT_PRC	**Port Reset Change (PRC).** Default = '0'. This flag is set to '1' due a '1' to '0' transition of Port Reset (PR). e.g. when any reset processing (Warm or Hot) on this port is complete. Note that this flag shall not be set to '1' if the reset processing was forced to terminate due to software clearing PP or PED to '0'. '0' = No change. '1' = Reset complete. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.5. Refer to section 4.19.2 for more information on change bit usage.

Bit(s)	Mnemonic	Name	Description
20		USB3_PORT_OCC	**Over-current Change (OCC).** Default = '0'. This bit shall be set to a '1' when there is a '0' to '1' or '1' to '0' transition of Over-current Active (OCA). Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage.
19		USB3_PORT_WRC	**Warm Port Reset Change (WRC).** Default = '0'. This bit is set when Warm Reset processing on this port completes. '0' = No change. '1' = Warm Reset complete. Note that this flag shall not be set to '1' if the Warm Reset processing was forced to terminate due to software clearing PP or PED to '0'. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.5.1. Refer to section 4.19.2 for more information on change bit usage. This bit only applies to USB3 protocol ports. For USB2 protocol ports it shall be RsvdZ.
18		USB3_PORT_PEC	**Port Enabled/Disabled Change (PEC).** Default = '0'. '1' = change in PED. '0' = No change. Note that this flag shall not be set if the PED transition was due to software setting PP to '0'. Software shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage. For a USB2 protocol port, this bit shall be set to '1' only when the port is disabled due to the appropriate conditions existing at the EOF2 point (refer to section 11.8.1 of the USB2 Specification for the definition of a Port Error). For a USB3 protocol port, this bit shall never be set to '1'.
17		USB3_PORT_CSC	**Connect Status Change (CSC) .** Default = '0'. '1' = Change in CCS. '0' = No change. This flag indicates a change has occurred in the port's Current Connect Status (CCS). Note that this flag shall not be set if the CCS transition was due to software setting PP to '0'. The xHC sets this bit to '1' for all changes to the port device connect status, even if system software has not cleared an existing Connect Status Change. For example, the insertion status changes twice before system software has cleared the changed condition, root hub hardware will be "setting" an already-set bit (i.e., the bit will remain '1'). Software

Bit(s)	Mnemonic	Name	Description
			shall clear this bit by writing a '1' to it. Refer to section 4.19.2 for more information on change bit usage.
16		USB3_PORT_LWS	**Port Link State Write Strobe (LWS).** Default = '0'. When this bit is set to '1' on a write reference to this register, this flag enables writes to the PLS field. When '0', write data in PLS field is ignored. Reads to this bit return '0'.
15:14		USB3_PORT_PIC	**Port Indicator Control (PIC).** Default = 0. Writing to these bits has no effect if the Port Indicators (PIND) bit in the HCCPARAMS register is a '0'. If PIND bit is a '1', then the bit encodings are: Value Meaning 0 Port indicators are off 1 Amber 2 Green 3 Undefined Refer to the USB2 Specification for a description on how these bits are to be used. This field is '0' if PP is '0'.
13:10		USB3_PORT_Port_Speed	**Port Speed (Port Speed).** Default = '0'. This field identifies the speed of the attached USB Device. This field is only relevant if a device is attached (CCS = '1') in all other cases this field shall indicate Undefined Speed. Value Meaning 0 Undefined Speed 1 Full-speed device attached 2 Low-speed device attached 3 High-speed device attached 4 SuperSpeed device attached 5-15 Reserved Note: Values the 1, 2 and 3 are exclusive to USB2 protocol ports and the value 4 is exclusive to USB3 protocol ports
9		USB3_PORT_PP	**Port Power (PP).** Default = '1'. This flag reflects a port's logical, power control state. Because host controllers can implement different methods of port power switching, this flag may or may not represent whether (VBus) power is actually applied to the port. When PP equals a '0' the port is nonfunctional and shall not report attaches, detaches, or

Bit(s)	Mnemonic	Name	Description
			Port Link State (PLS) changes. However, the port shall report over-current conditions when PP = '0' if PPC = '0'. 0 = This port is in the Powered-off state. 1 = This port is not in the Powered-off state. If the Port Power Control (PPC) flag in the HCCPARAMS register is '1', then xHC has port power control switches and this bit represents the current setting of the switch ('0' = off, '1' = on). If the Port Power Control (PPC) flag in the HCCPARAMS register is '0', then xHC does not have port power control switches and each port is hard wired to power, and not affected by this bit. When an over-current condition is detected on a powered port, the xHC shall transition the PP bit in each affected port from a '1' to '0' (removing power from the port). Refer to section 4.19.4 for more information.
8:5		USB3_PORT_PLS	**Port Link State (PLS).** Default = RxDetect ('5'). This field is used to power manage the port and reflects its current link state. When the port is in the Enabled state, system software may set the link U state by writing this field. System software may also write this field to force a Disabled to Disconnected state transition of the port. Write Value Description 0 The link shall transition to a U0 state from any of the U states. 2b USB2 protocol ports only. The link should transition to the U2 State. 3 The link shall transition to a U3 state from any of the U states. This action selectively suspends the device connected to this port. While the Port Link State = U3, the hub does not propagate downstream-directed traffic to this port, but the hub will respond to resume signaling from the port. 5 USB3 protocol ports only. If the port is in the Disabled state (PLS = Disabled, PP = 1), then the link shall transition to a RxDetect state and the port shall transition to the Disconnected state, else ignored. 1b,4,6-14 Ignored. 15 USB2 protocol ports only. If the port is in the U3 state (PLS = U3), then the link shall remain in the U3 state and the port shall transition to the U3Exit substate, else ignored. Refer to section 4.15.2 for more information. Note: The Port Link State Write Strobe (LWS) shall also

Bit(s)	Mnemonic	Name	Description
			be set to '1' to write this field. For USB2 protocol ports: Writing a value of '2' to this field shall request LPM, asserting L1 signaling on the USB2 bus. Software may read this field to determine if the transition to the U2 state was successful. Writing a value of '0' shall deassert L1 signaling on the USB. Writing a value of '1' shall have no affect. The U1 state shall never be reported by a USB2 protocol port. Read Value Meaning 0 Link is in the U0 State 1 Link is in the U1 State 2 Link is in the U2 State 3 Link is in the U3 State (Device Suspended) 4 Link is in the Disabled Statec 5 Link is in the RxDetect Stated 6 Link is in the Inactive Statee 7 Link is in the Polling State 8 Link is in the Recovery State 9 Link is in the Hot Reset State 10 Link is in the Compliance Mode State 11 Link is in the Test Modef State 12:13 Reserved 14: Link is in the Reset state (MTK) 15 Link is in the Resume Stateg This field is undefined if PP = '0'. Note: Transitions between different states are not reflected until the transition is complete. Refer to sections 4.15.2 and 4.23.5 for more information on the use of this field. Refer to the USB2 LPM ECR for more information on USB link power management operation. Refer to section 7.2 for supported USB protocols.
4		USB3_PORT_PR	**Port Reset (PR).** Default = '0'. '1' = Port Reset signaling is asserted. '0' = Port is not in Reset. When software writes a '1' to this bit (from a '0') the bus reset sequence is initiated; USB2 protocol ports shall execute the bus reset sequence as defined in the USB2 Spec. USB3 protocol ports shall execute the Hot Reset sequence as defined in the USB3 Spec. PR remains set until reset signaling is completed by the root hub. Note that software shall write a '1' to the this flag to transition a USB2 port from the Polling state to the Enabled state. Refer to sections 4.15.2.3 and 4.19.1.1. This flag is '0' if PP is '0'.
3		USB3_PORT_OCA	**Over-current Active (OCA).**

Bit(s)	Mnemonic	Name	Description
			Default = '0'. '1' = This port currently has an over-current condition. '0' = This port does not have an over-current condition. This bit shall automatically transition from a '1' to a '0' when the over-current condition is removed.
1		USB3_PORT_PED	**Port Enabled/Disabled (PED).** Default = '0'. '1' = Enabled. '0' = Disabled. Ports may only be enabled by the xHC. Software cannot enable a port by writing a '1' to this flag. A port may be disabled by software writing a '1' to this flag. This flag shall automatically be cleared to '0' by a disconnect event or other fault condition. Note that the bit status does not change until the port state actually changes. There may be a delay in disabling or enabling a port due to other host controller or bus events. When the port is disabled (PED = '0') downstream propagation of data is blocked on this port, except for reset. For USB2 protocol ports: When the port is in the Disabled state, software shall reset the port (PR = '1') to transition PED to '1' and the port to the Enabled state. For USB3 protocol ports: When the port is in the Polling state (after detecting an attach), the port shall automatically transition to the Enabled state and set PED to '1' upon the completion of successful link training. When the port is in the Disabled state, software shall write a '5' (RxDetect) to the PLS field to transition the port to the Disconnected state. Refer to section 4.19.1.2. PED shall automatically be cleared to '0' when PR is set to '1', and set to '1' when PR transitions from '1' to '0' after a successful reset. Refer to Port Reset (PR) bit for more information on how the PED bit is managed. Note that when software writes this bit to a '1', it shall also write a '0' to the PR bita. This flag is '0' if PP is '0'.
0		USB3_PORT_CCS	**Current Connect Status (CCS).** Default = '0'. '1' = Device is present on port. '0' = No device is present. This value reflects the current state of the port, and may not correspond directly to the event that caused the Connect Status Change (CSC) bit to be set to '1'. Refer to section 4.19.4 for more details on the Connect Status Change (CSC) assertion conditions. This flag is '0' if PP is '0'.

Bit(s)	Mnemonic	Name	Description

1A240424 USB3_PORT_PMSC USB3_PORT PM Status and Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																USB3_PORT_FLA
Type																RW
Reset																0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	USB3_PORT_U2_Timeout								USB3_PORT_U1_Timeout							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
16		USB3_PORT_FLA	**Force Link PM Accept (FLA).** Default = '0'. When this bit is set to '1', the port shall generate a Set Link Function LMP with the Force_LinkPM_Accept bit asserted. . This flag shall be set to '0' by the assertion of PR to '1' or when CCS = transitions from '0' to '1'. Writes to this flag have no affect if PP = '0'. The Set Link Function LMP is sent by the xHC to the device connected on this port when this bit transitions from '0' to '1'. Refer to Sections 8.4.1, 10.4.2.2 and 10.4.2.9 of the USB3 specification for more details. Improper use of the SS Force_LinkPM_Accept functionality can impact the performance of the link significantly. This bit shall only be used for compliance and testing purposes. Software shall ensure that there are no pending packets at the link level before setting this bit. This flag is '0' if PP is '0'.
15:8		USB3_PORT_U2_Timeout	**U2 Timeout.** Default = '0'. Timeout value for U2 inactivity timer. If equal to FFh, the port

Bit(s)	Mnemonic	Name	Description
			is disabled from initiating U2 entry. This field shall be set to '0' by the assertion of PR to '1'. Refer to section 4.19.4.1 for more information on U2 Timeout operation. The following are permissible values: Value Description 00h Zero (default) 01h 256 us 02h 512 us ... FEh 65.024 ms FFh Infinite A U2 Inactivity Timeout LMP shall be sent by the xHC to the device connected on this port when this field is written. Refer to Sections 8.4.3 and 10.4.2.10 of the USB3 specification for more details.
7:0		USB3_PORT_U1_Timeout	**U1 Timeout.** Default = '0'. Timeout value for U1 inactivity timer. If equal to FFh, the port is disabled from initiating U1 entry. This field shall be set to '0' by the assertion of PR to '1'. Refer to section 4.19.4.1 for more information on U1 Timeout operation. The following are permissible values: Value Description 00h Zero (default) 01h 1 us. 02h 2 us. ... 7Fh 127 us. 80h-FEh Reserved FFh Infinite

1A240428 USB3_PORT_LI USB3_PORT Link Info 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	USB3_PORT_Link_Error_Count															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Mnemonic	Name	Description
15:0		USB3_PORT_Link_Error_Count	**Link Error Count.** Default = '0'. This field returns the number of link errors detected by the port. This value shall be reset to '0' by the assertion of a Chip Hardware Reset, HCRST, when PR transitions from '1' to '0', or when CCS = transitions from '0' to '1'.

21 USB2.0 (in USB3.0)

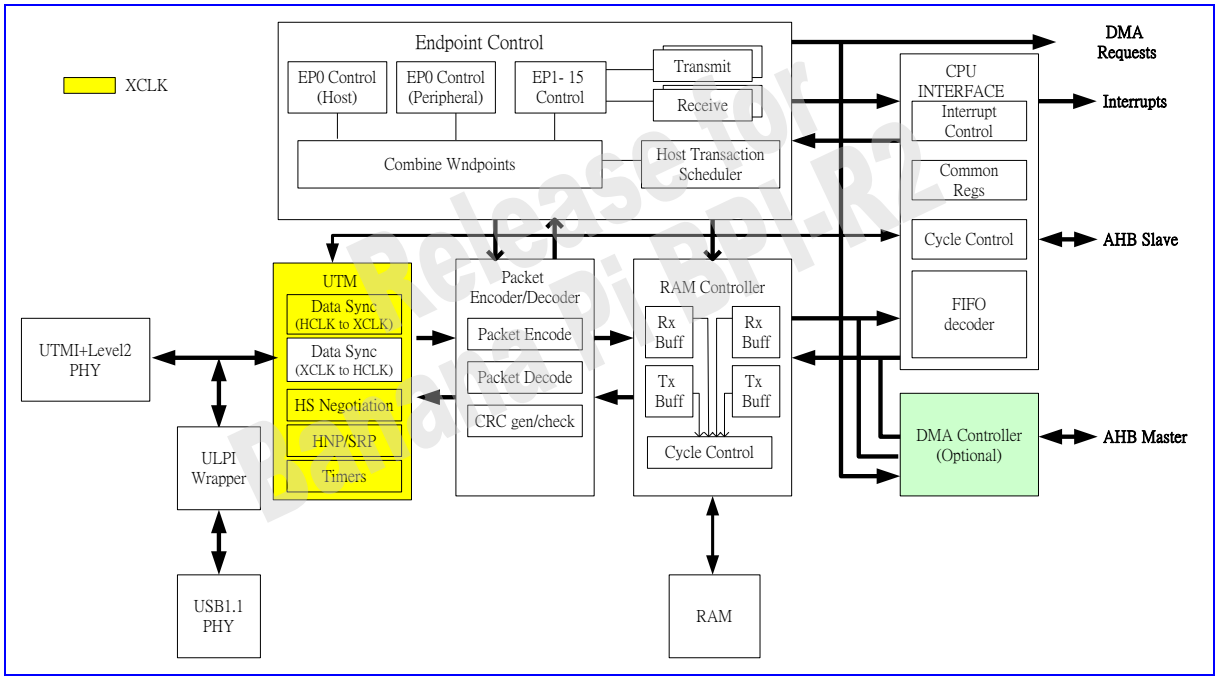
21.1 Introduction

Hereis the USB 2.0 OTG controller in USB3.0

21.2 Feature list

- USB2.0 with OTG
- 8 Tx and 8 Rx end point (exclude EP0)
- 4 Tx and 4 Rx USBQ

21.3 Block Diagram



21.4 Register Definition

Registers accessed using byte manipulation are marked in blue columns. Byte accessing registers can be accessed using word manipulation. Word accessing registers cannot be accessed using the byte manipulation.

Base address: (+11200000h)

Register address	Register name	Manipulation (Byte/Wo	Acronym

		rd)	
Common Registers			
USB + 0000h	Function address register	Byte	FADDR
USB + 0001h	Power management register	Byte	POWER
USB + 0002h	Tx interrupt status register	Byte	INTRTX
USB + 0004h	Rx interrupt status register	Byte	INTRRX
USB + 0006h	Tx interrupt enable register	Byte	INTRTXE
USB + 0008h	Rx interrupt enable register	Byte	INTRRXE
USB + 000Ah	Common USB interrupts register	Byte	INTRUSB
USB + 000Bh	Common USB interrupts enable register	Byte	INTRUSBE
USB + 000Ch	Frame number register	Byte	FRAME
USB + 000Eh	Endpoint selecting index register	Byte	INDEX
Indexed EndPoint CSR Region *n stands for endpoint number.* *For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint.* *MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0010h ~ USB + 001Fh	It maps to CSR EP0 ~ EPx depending on the INDEX register. For example, if INDEX is n, address 0010h ~ 001Fh are mapped to 0x(100+10*n)h ~ 0x(100+10*n+F)h.	Byte	Indexed CSR
USB + 0020h	USB endpoint 0 FIFO register	Byte	FIFO0
USB + 0020h +(n)*4 h	USB endpoint n FIFO register	Byte	FIFOn
OTG, Dynamic FIFO, Version Registers			
USB + 0060h	OTG device control register	Byte	DEVCTL
USB + 0061h	Power up counter register	Byte	PWRUPCNT
USB + 0062h	Tx FIFO size register	Byte	TXFIFOSZ
USB + 0063h	Rx FIFO size register	Byte	RXFIFOSZ
USB + 0064h	Tx FIFO address register	Byte	TXFIFOADD
USB + 0066h	Rx FIFO address register	Byte	RXFIFOADD
Hardware Configuration, Special Setting Registers			
USB + 007Ch	Time buffer available on HS transactions register	Byte	HS_EOF1
USB + 007Dh	Time buffer available on FS transactions register	Byte	FS_EOF1
USB + 007Eh	Time buffer available on LS transactions register	Byte	LS_EOF1
USB + 007Fh	Reset information register	Byte	RST_INFO
USB + 0080h	Rx data toggle set/status register	Word	RXTOG
USB + 0082h	Rx data toggle enable register	Word	RXTOGEN
USB + 0084h	Tx data toggle set/status register	Word	TXTOG
USB + 0086h	Tx data toggle enable register	Word	TXTOGEN
Level1 interrupt Control/Status registers			

USB + 00A0h	USB Level 1 interrupt status register	Byte	USB_L1INTS
USB + 00A4h	USB Level 1 interrupt unmask register	Byte	USB_L1INTM
USB + 00A8h	USB Level 1 interrupt polarity register	Byte	USB_L1INTP
USB + 00ACh	USB Level 1 interrupt control register	Byte	USB_L1INTC
Non-indexed EndPoint CSR Region *n stands for endpoint number.* *For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint.* *MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0102h	EP0 control status register	Byte	CSR0
USB + 0108h	EP0 received bytes register	Byte	COUNT0
USB + 010Bh	NAK limit register	Byte	NAKLIMT0
USB + 010Fh	Core configuration register	Byte	CONFIGDATA
USB + 0100h +(n)*10h	TXMAP register	Byte	TXMAP(n)
USB + 0102h +(n)*10h	Tx CSR register	Byte	TXCSR(n)
USB + 0104h +(n)*10h	RXMAP register	Byte	RXMAP(n)
USB + 0106h +(n)*10h	Rx CSR register	Byte	RXCSR(n)
USB + 0108h +(n)*10h	Rx Count register	Byte	RXCOUNT(n)
USB + 010Ah +(n)*10h	TxType register	Byte	TXTYPE(n)
USB + 010Bh +(n)*10h	TxInterval register	Byte	TXINTERVAL(n)
USB + 010Ch +(n)*10h	RxType register	Byte	RXTYPE(n)
USB + 010Dh +(n)*10h	RxInterval register	Byte	RXINTERVAL(n)
USB + 010Fh +(n)*10h	Configured FIFO size register	Byte	FIFOSIZE(n)
DMA Channels Control Registers *M stands for DMA channel number.* *For example, DMA channel 1's M = 1. Valid M = 1 ~ MaxDMAChannel.* *MaxDMAChannel is hardware configured and the maximum is 8.*			
USB + 0200h	DMA interrupt status register (word access only)	Word	DMA_INTR
USB + 0210h	DMA limiter register (word access only)	Word	DMA_LIMITER
USB + 0220h	DMA configuration register (word access only)	Word	DMA_CONFIG
USB + 0204h +(M-1)*10h	DMA channel M control register (word access only)	Word	DMA_CNTL_M
USB + 0208h	DMA channel M address register (word access only)	Word	DMA_ADDR_M

+(M-1)*10h			
USB + 020Ch +(M-1)*10h	DMA channel M byte count register (word access only)	Word	DMA_COUNT_M
EndPoint RX Packet Count Register *n stands for endpoint number. For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint. MaxEndPoint is hardware configured and the maximum is 15.*			
USB + 0300h +(n)*4h	EPn RxPktCount register	Word	EPnRXPKTCOUNT
Host/Hub Control Registers (Host mode only registers) *n stands for endpoint number. For example, endpoint 1's n = 1. Valid n = 1 ~ MaxEndPoint. MaxEndPoint is hardware configured and maximum is 15.*			
USB + 0480h +8*n h	Transmit endpoint n function address	Word	TXFUNCADDR
USB + 0482h +8*n h	Transmit endpoint n hub/port address	Word	TXHUBADDR
USB + 0484h +8*n h	Receive endpoint n function address	Word	RXFUNCADDR
USB + 0486h +8*n h	Receive endpoint n hub/port address	Word	RXHUBADDR

Please reference to chapter 12 for detail description.

22 ETHSYS Control

22.1 Description

This module collects sub-system's all global configuration registers. The hard-wire output is used to control the hardware (EX. CG enable or clock MUX selection…etc.). And the hard-wire input is used to collect the hardware status output (EX. ID or monitor output…etc.).

22.2 Feature List

- Sub-system global configuration
- Sub-system clock MUX and CG control
- Sub-system SW reset control
- Sub-system MEMO

22.3 Block Diagram

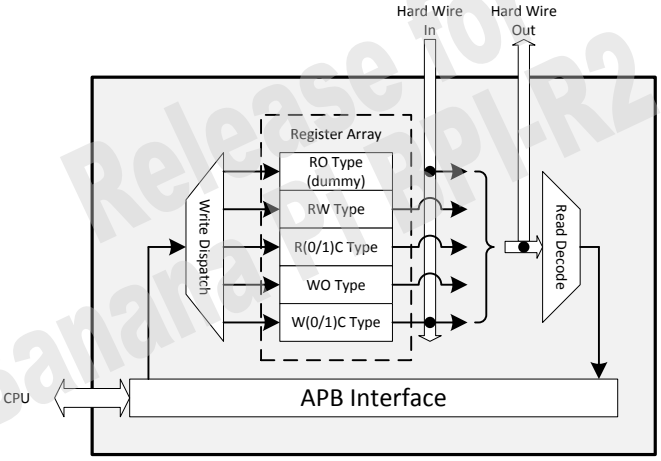


Figure 22-1 ETHSYS sub-system control

22.4 Register Definition

Module name: ETHDMASYS_SYSCTL Base address: (+1B000000h)

Address	Name	Width	Register Function
1B000000	ID0_3	32	ID ASCII Character 0-3
1B000004	ID4_7	32	ID ASCII Character 4-7
1B00000C	Revision_ID	32	Revision Identification
1B000014	SYSCFG1	32	System Configuration Register 1
1B00002C	CLKCFG0	32	Clock Configuration Register 0
1B000030	CLKCFG1	32	Clock Configuration Register 1
1B000034	RSTCTL	32	Reset Control Register

1B000000 **CHIPID0_3** **ID ASCII Character 0-3** **3637544D**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ID3								ID2							
Type	RO								RO							
Reset	0	0	1	1	0	1	1	0	0	0	1	1	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ID1								ID0							
Type	RO								RO							
Reset	0	1	0	1	0	1	0	0	0	1	0	0	1	1	0	1

Bit(s)	Name	Description
31:24	ID3	**ASCII Identification Character 3**
23:16	ID2	**ASCII Identification Character 2**
15:8	ID1	**ASCII Identification Character 1**
7:0	ID0	**ASCII Identification Character 0**

1B000004 **ID4_7** **ID ASCII Character 4-7** **20203332**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ID7								ID6							
Type	RO								RO							
Reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ID5								ID4							
Type	RO								RO							
Reset	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0

Bit(s)	Name	Description
31:24	ID7	**ASCII dentification Character 7**
23:16	ID6	**ASCII Identification Character 6**
15:8	ID5	**ASCII Identification Character 5**
7:0	ID4	**ASCII Identification Character 4**

1B000014 **SYSCFG1** **System Configuration Register 1** **00000014**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GE2_MODE		GE1_MODE								AXI_MST_SYNC_SEL		AXI_SLV_SYNC_SEL			AXI_SYN

1B000014 SYSCFG1 System Configuration Register 1 00000014

																C_MODE
Type	RW		RW								RW		RW			RW
Reset	0	0	0	0							0	1	0	1		0

Bit(s)	Name	Description
15:14	GE2_MODE	**Gigabit Port #2 Mode** Sets the interface mode on Gigabit port 2 0: RGMII Mode (10/100/1000 Mbps) 1: MII Mode (10/100 Mbps) 2: Reverse MII Mode (10/100 Mbps) 3: Reduced MII (10/100Mbps)
13:12	GE1_MODE	**Gigabit Port #1 Mode** Sets the interface mode on Gigabit port 1 0: RGMII Mode (10/100/1000 Mbps) 1: MII Mode (10/100 Mbps) 2: Reverse MII Mode (10/100 Mbps) 3: Reserved
5:4	AXI_MST_SYNC_SEL	**AXI aslice master direction AFIFO synchronizer selection** 0: 1 DFF SYNC 1: 2 DFF SYNC 2: 3 DFF SYNC 3: 3 DFF SYNC
3:2	AXI_SLV_SYNC_SEL	**AXI aslice slave direction AFIFO synchronizer selection** 0: 1 DFF SYNC 1: 2 DFF SYNC 2: 3 DFF SYNC 3: 3 DFF SYNC
0	AXI_SYNC_MODE	**AXI ASYNC slice configuration** 0: AXI bus in ASYNC 1: AXI bus in SYNC

1B00002C CLKCFG0 Clock Configuration Register 0 C4001880

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	OSC_1US_DIV															
Type	RW															
Reset	1	1	0	0	0	1										
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name				FE_CLK_SEL	TRGMII_CLK_SEL											
Type				RW	RW											
Reset				1	1											

Bit(s)	Name	Description
31:26	OSC_1US_DIV	**Oscillator 1 usec Divider** Sets the maximum for the reference clock counter for either a 20 MHz or 40 MHz external XTAL input. The count increments each 1usec (indicating 1

Bit(s)	Name	Description
		MHz), up to the maximum, before resetting to zero. This counts the frequency of an external XTAL. This count is used to output a 32 KHz frequency to the REFCLKo pin. 0: Automatically generates a 1 usec system tick regardless of whether XTAL frequency is 20 MHz or 40 MHz. 49: Default value for a reference 50 MHz clock. 39: Default value for an external 40 MHz XTAL. 19: Default value for an external 20 MHz XTAL. Others: Manual mode for tick generation.
12	FE_CLK_SEL	**Frame engine main clock selection** 1: from ETHPLL 500MHz 0: from ETHPLL 250MHz
11	TRGMII_CLK_SEL	**TRGMII Tx clock selection(when TRGMII @362.5MHz, ETHPLL need to fix @500MHz)** 1: from TRGMII 362.5MHz 0: from ETHPLL 250MHz

1B00003 0 **CLKCFG1** **Clock Configuration Register 1** **200249E0**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name																
Type																
Reset																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name								GP1_CLK_EN	GP2_CLK_EN							
Type								RW	RW							
Reset								1	1							

Bit(s)	Name	Description
8	GP1_CLK_EN	**Giga port 1 clock control** 1: Clock Enable 0: Clock Disable
7	GP2_CLK_EN	**Giga port 2 clock control** 1: Clock Enable 0: Clock Disable

1B000034 RSTCTL **Reset Control Register** **00000004**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PPE_RST								GMAC_RST							

Type	RW								RW							
Reset	0								0							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name										FE_RST						SYS_RST
Type										RW						W1C
Reset										0						0

Bit(s)	Name	Description
31	PPE_RST	**PPE reset control** 1: Reset Assert 0: Reset Deassert
23	GMAC_RST	**GMAC reset control** 1: Reset Assert 0: Reset Deassert
6	FE_RST	**FE reset control** 1: Reset Assert 0: Reset Deassert
0	SYS_RST	**Whole Sub-System Reset Control** 1: Whole System Reset 0: NA

23 Frame Engine

23.1 Introduction

Frame engine provides a DMA interface to transfer Ethernet packets between CPU and GMACs. Frame engine includes PDAM, QDMA, PSE, PPE, GDM, and CDM. The main features of these functions are list below.
Gigabit Ethernet is constructured by a single, compact component with two full-integrated Gigabit Media Access Control (GMAC) ports.The two GMAC ports are connected to the Frame Engine to perform the wire-speed NAT process between LAN and WAN domain. To communicate with the external GePHY, GMAC also provide the Serial Management Interface running up to 12.5MHz.

23.2 Feature List

23.2.1 PSE Features

- Wire-speed (1000 Mbps) Ethernet LAN/WAN NAT/NAPT routing
- Egress rate limiting/shaping
- Flow control for no-packet-loss guarantee
- Emulated multicast support for keep-alive (can mirror a Tx packet to CPU)
- IP/TCP/UDP Checksum offload
- IP/TCP/UDP Checksum Generation
- VLAN & PPPOE header insertion
- TCP Segmentation Offload
- Auto-Padding for sub-64 B packets

23.2.2 PDMA Features

- Supports 4 Tx descriptor rings and 4 Rx descriptor rings
- Scatter/Gather DMA
- Delayed interrupt
- Configurable 4/8/16/32 32-bit word burst length
- Support TSO(TCP Segmentation Offload)

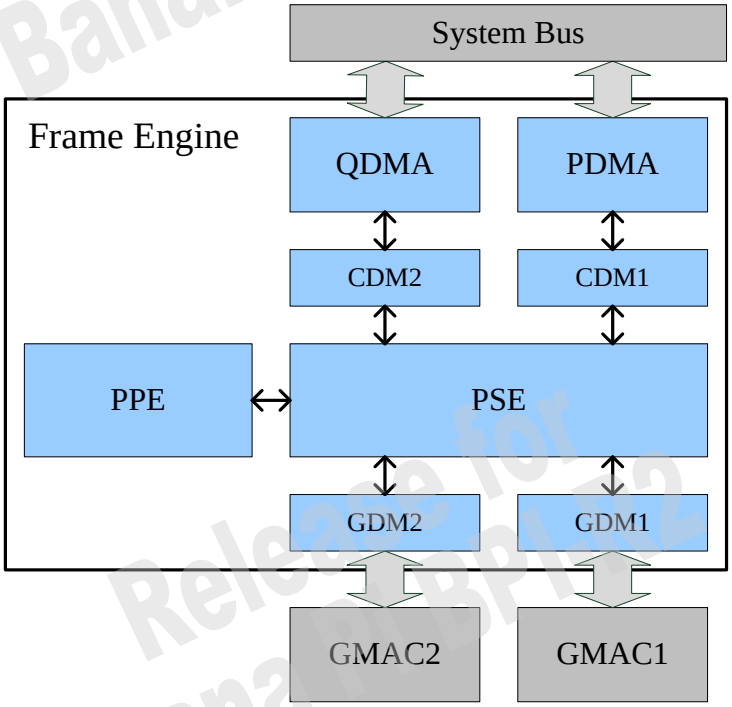
23.2.3 GMAC Features

- Follow IEEE Std. 802.3 MAC layer protocol
- VLAN support compliant with IEEE Std 802.1Q specifiction
- Supports 802.3x flow control for full duplex operation and backpressure flow control for half duplex operation.
- Energy Efficient Ethernet (EEE) support compliant with IEEE Std 802.3az
- Support Wake-on-Lan or Interrupt by the Magic packet
- Support the flexible PHY interfaces - MII/RMII/RGMII and TRGMII (GMAC1 only)

- Support Clause 22 and 45 MDIO access and GePHY auto-polling on SMI master.

23.3 Block Diagram

23.3.1 Frame Engine



23.4 Register Definition

Module name: FE Base address: (+1B100000h)

Address	Name	Width	Register Function
1B100000	FE_GLO_CFG	32	Frame Engine Global Configuration
1B100004	FE_RST_GLO	32	Frame Engine Global Reset
1B100008	FE_INT_STATUS	32	Frame Engine Interrupt Status
1B10000C	FE_INT_ENABLE	32	Frame Engine Interrupt Enable
1B100010	FE_FOE_TS_T	32	Frame Engine Time Stamp
1B100014	FE_IPV6_EXT	32	Frame Engine IPv6 Extension Header
1B100018	FE_RATE_COMP	32	Frame Engine Rate Limt Compensation
1B100020	FE_INT_GRP	32	Frame Engine Interrupt Group
1B100100	PSE_FQFC_CFG	32	PSE Free Queue Flow Control
1B100108	PSE_IQ_REV1	32	PSE Input Queue Reservation-I
1B10010C	PSE_IQ_REV2	32	PSE Input Queue Reservation-II
1B100110	PSE_IQ_STA1	32	PSE Input Queue Status-I
1B100114	PSE_IQ_STA2	32	PSE Input Queue Status-II

1B100118	PSE_OQ_STA1	32	PSE Output Queue Status-I
1B10011C	PSE_OQ_STA2	32	PSE Output Queue Status-II
1B100120	PSE_MIR_PORT	32	PSE Mirror Port
1B100130	FE_GDM_RXID1	32	FE GDM RXID Control 1
1B100134	FE_GDM_RXID2	32	FE GDM RXID Control 2
1B100400	CDMP_IG_CTRL	32	CDM Ingress Control
1B100404	CDMP_EG_CTRL	32	CDM Egress Control
1B100408	CDMP_PPE_GEN	32	CDM PPPoE Generation
1B100500	GDM1_IG_CTRL	32	GDM Ingress Control
1B100504	GDM1_EG_CTRL	32	GDM Egress Control
1B100508	GDM1_MAC_LSB	32	GDM MY_MAC Address LSB
1B10050C	GDM1_MAC_MSB	32	GDM MY_MAC Address MSB
1B100510	GDM1_VLAN_GEN	32	GDM VLAN Generation
1B100800	TX_BASE_PTR_0	32	TX Ring #0 Base Pointer
1B100804	TX_MAX_CNT_0	32	TX Ring #0 Maximum Count
1B100808	TX_CTX_IDX_0	32	TX Ring #0 CPU pointer
1B10080C	TX_DTX_IDX_0	32	TX Ring #0 DMA poitner
1B100810	TX_BASE_PTR_1	32	TX Ring #1 Base Pointer
1B100814	TX_MAX_CNT_1	32	TX Ring #1 Maximum Count
1B100818	TX_CTX_IDX_1	32	TX Ring #1 CPU pointer
1B10081C	TX_DTX_IDX_1	32	TX Ring #1 DMA poitner
1B100820	TX_BASE_PTR_2	32	TX Ring #2 Base Pointer
1B100824	TX_MAX_CNT_2	32	TX Ring #2 Maximum Count
1B100828	TX_CTX_IDX_2	32	TX Ring #2 CPU pointer
1B10082C	TX_DTX_IDX_2	32	TX Ring #2 DMA poitner
1B100830	TX_BASE_PTR_3	32	TX Ring #3 Base Pointer
1B100834	TX_MAX_CNT_3	32	TX Ring #3 Maximum Count
1B100838	TX_CTX_IDX_3	32	TX Ring #3 CPU pointer
1B10083C	TX_DTX_IDX_3	32	TX Ring #3 DMA poitner
1B100900	RX_BASE_PTR_0	32	RX Ring #0 Base Pointer
1B100904	RX_MAX_CNT_0	32	RX Ring #0 Maximum Count
1B100908	RX_CRX_IDX_0	32	RX Ring #0 CPU pointer
1B10090C	RX_DRX_IDX_0	32	RX Ring #0 DMA poitner
1B100910	RX_BASE_PTR_1	32	RX Ring #1 Base Pointer
1B100914	RX_MAX_CNT_1	32	RX Ring #1 Maximum Count
1B100918	RX_CRX_IDX_1	32	RX Ring #1 CPU pointer
1B10091C	RX_DRX_IDX_1	32	RX Ring #1 DMA poitner
1B100920	RX_BASE_PTR_2	32	RX Ring #2 Base Pointer
1B100924	RX_MAX_CNT_2	32	RX Ring #2 Maximum Count
1B100928	RX_CRX_IDX_2	32	RX Ring #2 CPU pointer
1B10092C	RX_DRX_IDX_2	32	RX Ring #2 DMA poitner
1B100930	RX_BASE_PTR_3	32	RX Ring #3 Base Pointer
1B100934	RX_MAX_CNT_3	32	RX Ring #3 Maximum Count

1B100938	RX_CRX_IDX_3	32	RX Ring #3 CPU pointer
1B10093C	RX_DRX_IDX_3	32	RX Ring #3 DMA poitner
1B100A00	PDMA_INFO	32	PDMA Information
1B100A04	PDMA_GLO_CFG	32	PDMA Global Configuration
1B100A08	PDMA_RST_IDX	32	PDMA Reset Index
1B100A0C	DELAY_INT_CFG	32	Delay Interrupt Configuration
1B100A10	FREEQ_THRES	32	Free Queue Threshold
1B100A20	INT_STATUS	32	Interrupt Status
1B100A28	INT_MASK	32	Interrupt Mask
1B100A40	PDMA_INT1_VEC_GRP0	32	PDMA Interrupt Status Group 0
1B100A44	PDMA_INT1_VEC_GRP1	32	PDMA Interrupt Status Group 1
1B100A48	PDMA_INT1_VEC_GRP2	32	PDMA Interrupt Status Group 2
1B100A50	PDMA_INT_GRP1	32	PDMA Interrupt Group 1 Control
1B100A54	PDMA_INT_GRP2	32	PDMA Interrupt Group 2 Control
1B101400	CDMQ_IG_CTRL	32	CDM VLAN Control
1B101404	CDMQ_EG_CTRL	32	CDM Egress Control
1B101408	CDMQ_PPP_GEN	32	CDM PPPoE Generation
1B101500	GDM2_IG_CTRL	32	GDM Ingress Control
1B101504	GDM2_EG_CTRL	32	GDM Egress Control
1B101508	GDM2_MAC_LSB	32	GDM MY_MAC Address LSB
1B10150C	GDM2_MAC_MSB	32	GDM MY_MAC Address MSB
1B101510	GDM2_VLAN_GEN	32	GDM VLAN Generation
1B101514	GDM2_FILTER_CRTL	32	GDM Filter Control
1B101518	GDM2_VIDF01	32	GDM VID Filter Control 01
1B10151C	GDM2_VIDF23	32	GDM VID Filter Control 23

1B101800	QTX_CFG_0	32	TX Queue #0 Configuration
1B101804	QTX_SCH_0	32	TX Queue #0 Schedule
1B101808	QTX_HEAD_0	32	TX Queue #0 Head Pointer
1B10180C	QTX_TAIL_0	32	TX Queue #0 Tail Pointer
1B101810	QTX_CFG_1	32	TX Queue #1 Configuration
1B101814	QTX_SCH_1	32	TX Queue #1 Schedule
1B101818	QTX_HEAD_1	32	TX Queue #1 Head Pointer
1B10181C	QTX_TAIL_1	32	TX Queue #1 Tail Pointer
1B101820	QTX_CFG_2	32	TX Queue #2 Configuration
1B101824	QTX_SCH_2	32	TX Queue #2 Schedule
1B101828	QTX_HEAD_2	32	TX Queue #2 Head Pointer
1B10182C	QTX_TAIL_2	32	TX Queue #2 Tail Pointer
1B101830	QTX_CFG_3	32	TX Queue #3 Configuration
1B101834	QTX_SCH_3	32	TX Queue #3 Schedule
1B101838	QTX_HEAD_3	32	TX Queue #3 Head Pointer

1B10183C	QTX_TAIL_3	32	TX Queue #3 Tail Pointer
1B101840	QTX_CFG_4	32	TX Queue #4 Configuration
1B101844	QTX_SCH_4	32	TX Queue #4 Schedule
1B101848	QTX_HEAD_4	32	TX Queue #4 Head Pointer
1B10184C	QTX_TAIL_4	32	TX Queue #4 Tail Pointer
1B101850	QTX_CFG_5	32	TX Queue #5 Configuration
1B101854	QTX_SCH_5	32	TX Queue #5 Schedule
1B101858	QTX_HEAD_5	32	TX Queue #5 Head Pointer
1B10185C	QTX_TAIL_5	32	TX Queue #5 Tail Pointer
1B101860	QTX_CFG_6	32	TX Queue #6 Configuration
1B101864	QTX_SCH_6	32	TX Queue #6 Schedule
1B101868	QTX_HEAD_6	32	TX Queue #6 Head Pointer
1B10186C	QTX_TAIL_6	32	TX Queue #6 Tail Pointer
1B101870	QTX_CFG_7	32	TX Queue #7 Configuration
1B101874	QTX_SCH_7	32	TX Queue #7 Schedule
1B101878	QTX_HEAD_7	32	TX Queue #7 Head Pointer
1B10187C	QTX_TAIL_7	32	TX Queue #7 Tail Pointer
1B101880	QTX_CFG_8	32	TX Queue #8 Configuration
1B101884	QTX_SCH_8	32	TX Queue #8 Schedule
1B101888	QTX_HEAD_8	32	TX Queue #8 Head Pointer
1B10188C	QTX_TAIL_8	32	TX Queue #8 Tail Pointer
1B101890	QTX_CFG_9	32	TX Queue #9 Configuration
1B101894	QTX_SCH_9	32	TX Queue #9 Schedule
1B101898	QTX_HEAD_9	32	TX Queue #9 Head Pointer
1B10189C	QTX_TAIL_9	32	TX Queue #9 Tail Pointer
1B1018A0	QTX_CFG_10	32	TX Queue #10 Configuration
1B1018A4	QTX_SCH_10	32	TX Queue #10 Schedule
1B1018A8	QTX_HEAD_10	32	TX Queue #10 Head Pointer
1B1018AC	QTX_TAIL_10	32	TX Queue #10 Tail Pointer
1B1018B0	QTX_CFG_11	32	TX Queue #11 Configuration
1B1018B4	QTX_SCH_11	32	TX Queue #11 Schedule
1B1018B8	QTX_HEAD_11	32	TX Queue #11 Head Pointer
1B1018BC	QTX_TAIL_11	32	TX Queue #11 Tail Pointer
1B1018C0	QTX_CFG_12	32	TX Queue #12 Configuration
1B1018C4	QTX_SCH_12	32	TX Queue #12 Schedule
1B1018C8	QTX_HEAD_12	32	TX Queue #12 Head Pointer
1B1018CC	QTX_TAIL_12	32	TX Queue #12 Tail Pointer
1B1018D0	QTX_CFG_13	32	TX Queue #13 Configuration
1B1018D4	QTX_SCH_13	32	TX Queue #13 Schedule
1B1018D8	QTX_HEAD_13	32	TX Queue #13 Head Pointer
1B1018DC	QTX_TAIL_13	32	TX Queue #13 Tail Pointer
1B1018E0	QTX_CFG_14	32	TX Queue #14 Configuration
1B1018E4	QTX_SCH_14	32	TX Queue #14 Schedule
1B1018E8	QTX_HEAD_14	32	TX Queue #14 Head Pointer
1B1018EC	QTX_TAIL_14	32	TX Queue #14 Tail Pointer
1B1018F0	QTX_CFG_15	32	TX Queue #15 Configuration
1B1018F4	QTX_SCH_15	32	TX Queue #15 Schedule

1B1018F8	QTX_HEAD_15	32	TX Queue #15 Head Pointer
1B1018FC	QTX_TAIL_15	32	TX Queue #15 Tail Pointer
1B101900	QRX_BASE_PTR_0	32	RX Ring #0 Base Pointer
1B101904	QRX_MAX_CNT_0	32	RX Ring #0 Maximum Count
1B101908	QRX_CRX_IDX_0	32	RX Ring #0 CPU pointer
1B10190C	QRX_DRX_IDX_0	32	RX Ring #0 DMA poitner
1B101910	QRX_BASE_PTR_1	32	RX Ring #1 Base Pointer
1B101914	QRX_MAX_CNT_1	32	RX Ring #1 Maximum Count
1B101918	QRX_CRX_IDX_1	32	RX Ring #1 CPU pointer
1B10191C	QRX_DRX_IDX_1	32	RX Ring #1 DMA poitner
1B101A00	QDMA_INFO	32	QDMA Information
1B101A04	QDMA_GLO_CFG	32	QDMA Global Configuration
1B101A08	QDMA_RST_IDX	32	QDMA Reset Index
1B101A0C	QDMA_DELAY_INT	32	Delay Interrupt Configuration
1B101A10	QDMA_FC_THRES	32	Flow ControlThreshold
1B101A14	QDMA_TX_SCH	32	TX Scheduler Rate Control
1B101A18	QDMA_INT_STS	32	Interrupt Status
1B101A1C	QDMA_INT_MASK	32	Interrupt Mask
1B101A20	QDMA_INT_GRP1	32	QDMA Interrupt Group 1 Control
1B101A24	QDMA_INT_GRP2	32	QDMA Interrupt Group 2 Control
1B101A2C	QDMA_DROP_PREC	32	QDMA Drop by FFA Percentage Control
1B101A40	QDMA_HRED1	32	QDMA HW RED Distribution - I
1B101A44	QDMA_HRED2	32	QDMA HW RED Distribution - II
1B101A48	QDMA_SRED1	32	QDMA SW RED Distribution - I
1B101A4C	QDMA_SRED2	32	QDMA SW RED Distribution - II
1B101A50	QTX_Q0MIN_BK	32	QDMA Queue #0 Min. Bucket
1B101A54	QTX_Q1MIN_BK	32	QDMA Queue #1 Min. Bucket
1B101A58	QTX_Q2MIN_BK	32	QDMA Queue #2 Min. Bucket
1B101A5C	QTX_Q3MIN_BK	32	QDMA Queue #3 Min. Bucket
1B101A60	QTX_Q0MAX_BK	32	QDMA Queue #0 Max. Bucket
1B101A64	QTX_Q1MAX_BK	32	QDMA Queue #1 Max. Bucket
1B101A68	QTX_Q2MAX_BK	32	QDMA Queue #2 Max. Bucket
1B101A6C	QTX_Q3MAX_BK	32	QDMA Queue #3 Max. Bucket
1B101A70	QDMA_INT_STS_G0	32	QDMA Interrupt Status Group 0
1B101A74	QDMA_INT_STS_G1	32	QDMA Interrupt Status Group 1
1B101A78	QDMA_INT_STS_G2	32	QDMA Interrupt Status Group 2
1B101AD0	QTX_MIB_PCNT	32	TX Queue MIB Forward Packet Count
1B101AD4	QTX_MIB_DPCN	32	TX Queue MIB Dropped Packet Count

	T		
1B101B00	QTX_CTX_PTR	32	TX Forward CPU Pointer
1B101B04	QTX_DTX_PTR	32	TX Forward DMA Pointer
1B101B08	QTX_FWD_CNT	32	TX Forward DMA Counter
1B101B10	QTX_CRX_PTR	32	TX Release CPU Pointer
1B101B14	QTX_DRX_PTR	32	TX Release DMA Pointer
1B101B18	QTX_RLS_CNT	32	TX Release DMA Counter
1B101B20	QDMA_FQ_HEAD	32	Free Page Head Pointer
1B101B24	QDMA_FQ_TAIL	32	Free Page Tail Pointer
1B101B28	QDMA_FQ_CNT	32	Free Page Counter
1B101B2C	QDMA_FQ_BLEN	32	Free Page Buffer Length
1B101B80	QDMA_RATE_EXP0	32	QDMA Scheduler Rate Exponent 0
1B101B84	QDMA_RATE_EXP1	32	QDMA Scheduler Rate Exponent 1
1B101B88	QDMA_RATE_EXP2	32	QDMA Scheduler Rate Exponent 2
1B101B8C	QDMA_RATE_EXP3	32	QDMA Scheduler Rate Exponent 3
1B101B90	QDMA_RATE_EXP4	32	QDMA Scheduler Rate Exponent 4
1B101B94	QDMA_RATE_EXP5	32	QDMA Scheduler Rate Exponent 5

Module name: GMAC Base address: (+1b110000h)

Address	Name	Width	Register Function
1b110000	MAC_PPSC	32	PHY Polling and SMI Master Control
1b110004	MAC_PIAC	32	PHY Indirect Access Control
1b110008	MAC_GPC	32	GIGA MAC Port Control
1b110100	MAC_P1_MCR	32	Port 1 MAC Control
1b110104	MAC_P1_EEE	32	Port 1 EEE Control
1b110108	MAC_P1_SR	32	Port 1 MAC Status
1b110200	MAC_P2_MCR	32	Port 2 MAC Control
1b110204	MAC_P2_EEE	32	Port 2 EEE Control
1b110208	MAC_P2_SR	32	Port 2 MAC Status
1b110210	MAC_P2_WOL	32	Port 2 WOL

1B100000 FE_GLO_CFG Frame Engine Global Configuration 810000B0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	EXT_TPID															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LINK_DWN								L2_SPACE				INF_SPACE			
Type	RW								RW				RW			
Reset	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0

Bit(s)	Name	Description
31:16	EXT_TPID	**Extended VLAN TPID** User-defined TPID except for 0x8100, 0x88a8. [note] This TPID field is used by CDM/GDM RX direction.
15:8	LINK_DWN	**Disable PSE Port** When PSE port is not valid on the Frame Engine, user is suggested to disable the corresponding port.
7:4	L2_SPACE	**L2 space** The space unit is 8 bytes.
3:0	INF_SPACE	**PKT_INFO space** The space unit is 8 bytes. [Note1*] Per received packet has already occupied 8 bytes(1 unit) by default. The actual PKT_INFO space is equal to (INF_SPACE + 1) [Note2*] (L2_SPACE + INF_SPACE + 1) <= 0xF

1B100004 FE_RST_GLO Frame Engine Global Reset 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0[28:13]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV0[12:0]													PSE_RAM	PSE_MEM_EN	PSE_RESET
Type	DC													RW	RW	W1
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:3	RESV0	**Reserved**
2	PSE_RAM	**PSE accessable RAM mode** [note*1] when this bit is reset, PSE packet memory is only accessable by Pbus (0xA000~0xDFFF). [note*2] when this bit is set, PSE packet memory is only accessable by Pbus (0x8000~0xDFFF) 0: Only 8KB PSE memory is left for iNIC initialization. 1: Make the whole PSE memory be used for Pbus access
1	PSE_MEM_EN	**PSE access enable** 0: disable 1: Enable PSE memory is accessable by PBUs
0	PSE_RESET	**PSE soft reset (self-cleared)** [note*1] Whenever SW set this bit, PSE will be initialized again and enter the normal mode. 0: invalid 1: Write 1 to reset PSE

1B100008 FE_INT_STATUS Frame Engine Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PPE_AF	REV0	GDM2_AF	GDM1_AF	REV1		MAC2_LINK	MAC1_LINK	GDM2_ERR	GDM2_CRC	GDM1_ERR	GDM1_CRC	RFIFO_UF	RFIFO_OV	INFIFO_GET_ER	AFIFO_GET_ERR

															R	
Type	W1C	DC	W1C	W1C	DC		W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV2	CDM_TSO_ALIGN	CDM_TSO_ILLEGAL	CDM_TSO_FAIL	REV3		PSE_DROP	FQ_EMPTY	PSE_FC_ON							
Type	DC	W1C	W1C	W1C	DC		W1C	W1C	W1C							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	PPE_AF	**MIB counter is almost full on PPE**
30	REV0	**Reserved**
29	GDM2_AF	**MIB counter is almost full on GDM2**
28	GDM1_AF	**MIB counter is almost full on GDM1**
27:26	REV1	**Reserved**
25	MAC2_LINK	**Link Status Change on MAC2**
24	MAC1_LINK	**Link Status Change on MAC1**
23	GDM2_ERR	**Packet error (checksum, length, ..etc) on GDM2** Includign FIFO overflow, checksum error, undersize, oversize
22	GDM2_CRC	**Packet CRC error on GDM2**
21	GDM1_ERR	**Packet error (checksum, length, ..etc) on GDM1** Includign FIFO overflow, checksum error, undersize, oversize
20	GDM1_CRC	**Packet CRC error on GDM1**
19	RFIFO_UF	**Ring FIFO Underrun Defect**
18	RFIFO_OV	**Ring FIFO Overflow Defect**
17	INFIFO_GET_ERR	**IN FIFO Get Defect**
16	AFIFO_GET_ERR	**ASYNC FIFO Get Defect**
15	REV2	**Reserved**
14	CDM_TSO_ALIGN	**Detect EOF Alignment**
13	CDM_TSO_ILLEGAL	**Ignored illegal packet on CDM TSO**
12	CDM_TSO_FAIL	**TSO Fail on CDM**
11:10	REV3	**Reserved**
9	PSE_DROP	**Packet drop by PSE**
8	FQ_EMPTY	**PSE free queue is empty**
7:0	PSE_FC_ON	**PSE PORT flow control is asserted** PSE port[n] enables the flow control.

1B10000C FE_INT_ENABLE Frame Engine Interrupt Enable 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PPE_AF	REV0	GDM2_AF	GDM1_AF	REV1		MAC2_LINK	MAC1_LINK	GDM2_ERR	GDM2_CRC	GDM1_ERR	GDM1_CRC	RFIFO_UF	RFIFO_OV	INFIFO_GET_ERR	AFIFO_GET_ERR
Type	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV2	CDM_	CDM_	CDM_	REV3		PSE_	FQ_E	PSE_FC_ON							

		TSO_ALIGN	TSO_ILLEGAL	TSO_FAIL			DROP	MPTY								
Type	RW	RW	RW	RW	RW		RW	RW	RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	PPE_AF	
30	REV0	
29	GDM2_AF	
28	GDM1_AF	
27:26	REV1	
25	MAC2_LINK	
24	MAC1_LINK	
23	GDM2_ERR	
22	GDM2_CRC	
21	GDM1_ERR	
20	GDM1_CRC	
19	RFIFO_UF	
18	RFIFO_OV	
17	INFIFO_GET_ERR	
16	AFIFO_GET_ERR	
15	REV2	
14	CDM_TSO_ALIGN	
13	CDM_TSO_ILLEGAL	
12	CDM_TSO_FAIL	
11:10	REV3	
9	PSE_DROP	
8	FQ_EMPTY	
7:0	PSE_FC_ON	

1B100010 FE_FOE_TS_T Frame Engine Time Stamp 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FOE_TS_T															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	FOE_TS_T	**Time stamp** Time stamp unit is 1 sec. [Note] This timer is drived by free-running 125MHz clock

1B100014 FE_IPV6_EXT Frame Engine IPv6 Extension Header 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	IP6_EXT3								IP6_EXT2							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	IP6_EXT1								IP6_EXT0							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	IP6_EXT3	**IPv6 extension header #3**
23:16	IP6_EXT2	**IPv6 extension header #2**
15:8	IP6_EXT1	**IPv6 extension header #1**
7:0	IP6_EXT0	**IPv6 extension header #0**

1B100018 FE_RATE_COMP Frame Engine Rate Limt Compensation 00001818

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	DMA_RATE_MINUS	DMA_RATE_BYTE							PSE_RATE_MINUS	PSE_RATE_BYTE						
Type	RW	RW							RW	RW						
Reset	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0	0

Bit(s)	Name	Description
31:16	REV0	**Reserved**
15	DMA_RATE_MINUS	1'b0: a specific byte count is added to the frame length according to FE_GLO_CFG.RATE_BYTE 1'b1: a specific byte count is subtracted from the incoming frame length.
14:8	DMA_RATE_BYTE	**Byte number for rate control** The byte number should be subtraced while calculating the rate limit.
7	PSE_RATE_MINUS	1'b0: a specific byte count is added to the frame length according to FE_GLO_CFG.RATE_BYTE 1'b1: a specific byte count is subtracted from the incoming frame length.
6:0	PSE_RATE_BYTE	**Byte number for rate control** The byte number should be subtraced while calculating the rate limit.

1B100020 FE_INT_GRP Frame Engine Interrupt Group 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_G2_ASG				QDMA_INT_G1_ASG				QDMA_INT_G0_ASG				PDMA_INT_G2_ASG			
Type	RW				RW				RW				RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDMA_INT_G1_ASG				PDMA_INT_G0_ASG								FE_MISC_INT_ASG			

Type	RW				RW								RW			
Reset	0	0	0	0	0	0	0	0					0	0	0	0

Bit(s)	Name	Description
31:28	QDMA_INT_G2_ASG	**QDMA group 2 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
27:24	QDMA_INT_G1_ASG	**QDMA group 1 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
23:20	QDMA_INT_G0_ASG	**QDMA group 0 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
19:16	PDMA_INT_G2_ASG	**PDMA group 2 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
15:12	PDMA_INT_G1_ASG	**PDMA group 1 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
11:8	PDMA_INT_G0_ASG	**PDMA group 0 interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved
3:0	FE_MISC_INT_ASG	**Frame engine misc. interrupt assignment** 4'h0: Assign to fe_int[0] 4'h1: Assign to fe_int[1] 4'h2: Assign to fe_int[2] Others: Reserved

1B100100 PSE_FQFC_CFG PSE Free Queue Flow Control 6060000C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FQ_PCNT								FQ_MAX_PCNT							
Type	RO								RW							
Reset	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PPE_VIQ								PPE_LTH							
Type	RO								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0

Bit(s)	Name	Description
31:24	FQ_PCNT	**Free queue page count**

Bit(s)	Name	Description
23:16	FQ_MAX_PCNT	**Maximum free Q page count.** Please reset PSE after re-programming this register
15:8	PPE_VIQ	**PPE virtual input queue depth** This value is equal to SUM(P[n]_IQ_PCNT - P[n]_IQ_RES).
7:0	PPE_LTH	**Page count low threshold for PPE port** When the page count of PPE port is lower than PPE_LTH, PSE will disable flow control.

1B100108 PSE_IQ_REV1 PSE Input Queue Reservation-I 000C0C08

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								P2_IQ_RES							
Type	DC								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P1_IQ_RES								P0_IQ_RES							
Type	RW								RW							
Reset	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:16	P2_IQ_RES	**Port 2 virtual input Q reservation** [note*] After the flow control of GDM1/GDM2 is asserted, it can be released when P[n]_IQ_PCNT is lower than (P[n]_IQ_RES/2).
15:8	P1_IQ_RES	**Port 1 virtual input Q reservation** [note*] After the flow control of GDM1/GDM2 is asserted, it can be released when P[n]_IQ_PCNT is lower than (P[n]_IQ_RES/2).
7:0	P0_IQ_RES	**Port 0 virtual input Q reservation** When (PPE_VIQ > PPE_HTH) and P[n]_IQ_PCNT > P[n]_IQ_RES), the corresponding port will enable the flow control.

1B10010C PSE_IQ_REV2 PSE Input Queue Reservation-II 10080818

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FREE_DROP								P6_IQ_RES							
Type	RW								RW							
Reset	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P5_IQ_RES								PPE_HTH							
Type	RW								RW							
Reset	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0

Bit(s)	Name	Description
31:24	FREE_DROP	**Free Buffer Drop Threshold**
23:16	P6_IQ_RES	**Port 6 virtual input Q reservation**
15:8	P5_IQ_RES	**Port 5 virtual input Q reservation**
7:0	PPE_HTH	**Page count reservation for PPE port** When PPE_VIQ is bigger than PPE_HTH, PSE will check per port input queue depth to enable flow control.

1B100110 PSE_IQ_STA1 PSE Input Queue Status-I 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								P2_IQ_PCNT							
Type	DC								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P1_IQ_PCNT								P0_IQ_PCNT							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:16	P2_IQ_PCNT	**Port 2 virtual input Q page count**
15:8	P1_IQ_PCNT	**Port 1 virtual input Q page count**
7:0	P0_IQ_PCNT	**Port 0 virtual input Q page count**

1B100114 PSE_IQ_STA2 PSE Input Queue Status-II 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								P6_IQ_PCNT							
Type	DC								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P5_IQ_PCNT								REV1							
Type	RO								DC							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:16	P6_IQ_PCNT	**Port 6 virtual input Q page count**
15:8	P5_IQ_PCNT	**Port 5 virtual input Q page count**
7:0	REV1	**Reserved**

1B100118 PSE_OQ_STA1 PSE Output Queue Status-I 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								P2_OQ_PCNT							
Type	DC								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P1_OQ_PCNT								P0_OQ_PCNT							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:16	P2_OQ_PCNT	**Port 2 output Q page count**
15:8	P1_OQ_PCNT	**Port 1 output Q page count**
7:0	P0_OQ_PCNT	**Port 0 output Q page count**

1B10011C PSE_OQ_STA2 PSE Output Queue Status-II 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								REV1							
Type	DC								DC							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	P5_IQ_PCNT								PPE_OQ_PCNT							
Type	RO								RO							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:16	REV1	**Reserved**
15:8	P5_IQ_PCNT	**Port 5 output Q page count**
7:0	PPE_OQ_PCNT	**PPE output Q page count**

1B100120 PSE_MIR_PORT PSE Mirror Port 00777777

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0								P5_MIR_PX				REV1			
Type	DC								RW				DC			
Reset	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV2				P2_MIR_PX				P1_MIR_PX				P0_MIR_PX			
Type	DC				RW				RW				RW			
Reset	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1

Bit(s)	Name	Description
31:24	REV0	**Reserved**
23:20	P5_MIR_PX	**Port 5 Mirror Port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
19:16	REV1	**Reserved**
15:12	REV2	**Reserved**
11:8	P2_MIR_PX	**Port 2 Mirror Port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved

Bit(s)	Name	Description
7:4	P1_MIR_PX	**Port 1 Mirror Port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
3:0	P0_MIR_PX	**Port 0 Mirror Port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved

1B100130 FE_GDM_RXID1 FE GDM RXID Control 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM_VLAN_PRI7_RXID_SEL		GDM_VLAN_PRI6_RXID_SEL		GDM_VLAN_PRI5_RXID_SEL		GDM_VLAN_PRI4_RXID_SEL		GDM_VLAN_PRI3_RXID_SEL		GDM_VLAN_PRI2_RXID_SEL		GDM_VLAN_PRI1_RXID_SEL		GDM_VLAN_PRI0_RXID_SEL	
Type	RW		RW		RW		RW		RW		RW		RW		RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV										GDM_TCP_ACK_RXID_SEL		GDM_TCP_ACK_WZPC	GDM_RXID_PRI_SEL		
Type	DC										RW		RW	RW		
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	GDM_VLAN_PRI7_RXID_SEL	**GDM RXID Select for VLAN Priority = 7** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
29:28	GDM_VLAN_PRI6_RXID_SEL	**GDM RXID Select for VLAN Priority = 6** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
27:26	GDM_VLAN_PRI5_RXID_SEL	**GDM RXID Select for VLAN Priority = 5** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
25:24	GDM_VLAN_PRI4_RXID_SEL	**GDM RXID Select for VLAN Priority = 4** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2)

Bit(s)	Name	Description
		2'b11: Select PDMA RX Ring 3 (RXID=3)
23:22	GDM_VLAN_PRI3_RX ID_SEL	**GDM RXID Select for VLAN Priority = 3** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
21:20	GDM_VLAN_PRI2_RX ID_SEL	**GDM RXID Select for VLAN Priority = 2** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
19:18	GDM_VLAN_PRI1_RX ID_SEL	**GDM RXID Select for VLAN Priority = 1** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
17:16	GDM_VLAN_PRI0_RX ID_SEL	**GDM RXID Select for VLAN Priority = 0** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
15:6	REV	
5:4	GDM_TCP_ACK_RXI D_SEL	**GDM RXID Select for TCP ACK** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
3	GDM_TCP_ACK_WZ PC	**GDM TCP ACK with zero payload check (L3 payload = L4 header)**
2:0	GDM_RXID_PRI_SEL	**GDM RXID Select Priority Setting** Select Priority by PID/STAG, VLAD_PRI, and TCP_ACK 3'b000: PID/STAG 3'b001: VLAN_PRI --> PID/STAG 3'b010: TCP_ACK --> PID/STAG 3'b011: VLAN_PRI --> TCP_ACK --> PID/STAG 3'b100: TCP_ACK --> VLAN_PRI --> PID/STAG

1B100134 FE_GDM_RXID2 FE GDM RXID Control 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM_STAG7_RXID_SEL		GDM_STAG6_RXID_SEL		GDM_STAG5_RXID_SEL		GDM_STAG4_RXID_SEL		GDM_STAG3_RXID_SEL		GDM_STAG2_RXID_SEL		GDM_STAG1_RXID_SEL		GDM_STAG0_RXID_SEL	
Type	RW		RW		RW		RW		RW		RW		RW		RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV												GDM_PID2_RXID_SEL		GDM_PID1_RXID_SEL	
Type	DC												RW		RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	GDM_STAG7_RXID_ SEL	**GDM RXID Select for STAG = 7** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1)

Bit(s)	Name	Description
		2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
29:28	GDM_STAG6_RXID_SEL	**GDM RXID Select for STAG = 6** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
27:26	GDM_STAG5_RXID_SEL	**GDM RXID Select for STAG = 5** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
25:24	GDM_STAG4_RXID_SEL	**GDM RXID Select for STAG = 4** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
23:22	GDM_STAG3_RXID_SEL	**GDM RXID Select for STAG = 3** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
21:20	GDM_STAG2_RXID_SEL	**GDM RXID Select for STAG = 2** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
19:18	GDM_STAG1_RXID_SEL	**GDM RXID Select for STAG = 1** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
17:16	GDM_STAG0_RXID_SEL	**GDM RXID Select for STAG = 0** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
15:4	REV	
3:2	GDM_PID2_RXID_SEL	**GDM RXID Select for GDM2 (Port ID = 2)** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)
1:0	GDM_PID1_RXID_SEL	**GDM RXID Select for GDM1 (Port ID = 1)** 2'b00: Select PDMA RX Ring 0 (RXID=0) 2'b01: Select PDMA RX Ring 1 (RXID=1) 2'b10: Select PDMA RX Ring 2 (RXID=2) 2'b11: Select PDMA RX Ring 3 (RXID=3)

1B100400 CDMP_IG_CTRLCDM Ingress Control 81000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CDM_TPID															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0															STAG_EN
Type	DC															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	CDM_TPID	**Inserted VLAN TPID**
15:1	REV0	**Reserved**
0	STAG_EN	**Special tag indication** Indicate that the received packets is carrying the special tag from CPU . 0: no special tag inserted. 1: The first 2-byte after Source Address is the special tag.

1B100404 CDMP_EG_CTRL CDM Egress Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0[30:15]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0[14:0]															UNTAG_EN
Type	DC															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	REV0	**Reserved**
0	UNTAG_EN	**VLAN untag** Untag the egress packets which are transmitted from CDM to CPU 0: disable 1: The first 4-bytes VLAN tag after Source Address will be untagged

1B100408 CDMP_PPE_GEN CDM PPPoE Generation 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0															PPP_INS
Type	DC															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SESS_ID															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:17	REV0	**Reserved**
16	PPP_INS	**PPPoE Header Insertion**
15:0	SESS_ID	**PPPoE Session ID**

1B100500 GDM1_IG_CTRL GDM Ingress Control 00717777

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0						INSV_EN	STAG_EN	REV1	GDM_ICS_EN	GDM_TCS_EN	GDM_UCS_EN	DROP_256B	REV2		STRP_CRC
Type	DC						RW	RW	DC	RW	RW	RW	RW	DC		RW
Reset	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MYMAC_DP				BC_DP				MC_DP				UN_DP			
Type	RW				RW				RW				RW			
Reset	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1

Bit(s)	Name	Description
31:26	REV0	**Reserved**
25	INSV_EN	**VLAN insertion** Insert Port VID on the received packets on the corresponding GDM port. 0: disable 1: Insert 4-byte VLAN tag after Source Address
24	STAG_EN	**Special tag indication** Indicate that the received packets is carrying the special tag from GDM port. 0: no special tag inserted. 1: The first 2-byte after Source Address is the special tag.
23	REV1	**Reserved**
22	GDM_ICS_EN	**IPv4 header checksum error drop** 0: checksum error status reported on RX descriptor (IP4F) 1: checksum error packet will be dropped
21	GDM_TCS_EN	**TCP checksum error drop** 0: checksum error status reported on RX descriptor (L4F) 1: checksum error packet will be dropped
20	GDM_UCS_EN	**UDP checksum error drop** 0: checksum error status reported on RX descriptor (L4F) 1: checksum error packet will be dropped
19	DROP_256B	**A special mode to drop packets with payload > 256-bytes** 0: Drop packets according to the standard Ethernet frame length limitation. 1: Dopr packets with payloa > 256 bytes
18:17	REV2	**Reserved**
16	STRP_CRC	**GDM RX CRC Stripping** 1'b: Disable GDM RX CRC stripping 1'b1: Enable GDM RX CRC stripping
15:12	MYMAC_DP	**MY_MAC frames destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
11:8	BC_DP	**Broadcast frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved

Bit(s)	Name	Description
		4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
7:4	MC_DP	**Multicast frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
3:0	UN_DP	**Other frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved

1B100504 GDM1_EG_CTRL GDM Egress Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0	UNTAG_EN	DIS_PAD	DIS_CRC	REV1			SHPR_EN	BK_SIZE							
Type	DC	RW	RW	RW	DC			RW	RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TK_TICK	REV2	TK_RATE													
Type	RW	DC	RW													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	REV0	
30	UNTAG_EN	**VLAN untagging** Untagg the egress packets which are transmitted from GDM. 0: disable 1: The first 4-bytes VLAN tag after Source Address will be untagged
29	DIS_PAD	**GMAC Tx padding fucntion** 1'b0: Enable GMAC Tx padding 1'b1: Disable GMAC Tx padding
28	DIS_CRC	**GMAC TX CRC generation** 1'b0: Enable GMAC Tx CRC generation 1'b1: Disable GMAC Tx CRC generation
27:25	REV1	**Reserved**
24	SHPR_EN	**GDM output shaper enable**

Bit(s)	Name	Description
		1'b0: Disable 1'b1: Enable
23:16	BK_SIZE	**GDM output shaper bucket size.** This unit is 1K-byte
15	TK_TICK	**GDM shaper token period** 1'b0: GDM shaper add token every 1ms 1'b1: GDM shaper add token every 20us
14	REV2	**Reserved**
13:0	TK_RATE	**GDM output shaper token rate** The unit is 8-Byte/ms or 8-Byte/20us according to TK_TICK.

1B100508 GDM1_MAC_LSB GDM MY_MAC Address LSB 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAC_ADR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAC_ADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAC_ADR	**MY_MAC Address [31:0]**

1B10050C GDM1_MAC_MSB GDM MY_MAC Address MSB 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAC_ADR															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	MAC_ADR	**MY_MAC Address [47:32]**

1B100510 GDM1_VLAN_GEN GDM VLAN Generation 81000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM_TPID															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GDM_PRI			GDM_CFI	GDM_VID											
Type	RW			RW	RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	GDM_TPID	**Inserted VLAN TPID**
15:13	GDM_PRI	**Inserted PRI**
12	GDM_CFI	**Inserted CFI**
11:0	GDM_VID	**Inserted VLAN ID**

1B100800 TX_BASE_PTR_0 TX Ring #0 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

1B100804 TX_MAX_CNT_0 TX Ring #0 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

1B100808 TX_CTX_IDX_0 TX Ring #0 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_CTX_IDX											
Type	DC				RW											

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1B10080C TX_DTX_IDX_0 TX Ring #0 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_DTX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

1B100810 TX_BASE_PTR_1 TX Ring #1 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

1B100814 TX_MAX_CNT_1 TX Ring #1 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

1B100818 TX_CTX_IDX_1 TX Ring #1 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_CTX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1B10081C TX_DTX_IDX_1 TX Ring #1 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_DTX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

1B100820 TX_BASE_PTR_2 TX Ring #2 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

1B100824 TX_MAX_CNT_2 TX Ring #2 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															

Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

1B100828 TX_CTX_IDX_2 TX Ring #2 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_CTX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1B10082C TX_DTX_IDX_2 TX Ring #2 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_DTX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

1B100830 TX_BASE_PTR_3 TX Ring #3 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_BASE_PTR[15:0]															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_BASE_PTR	**Point to the base address of TX Ring #0 (4-DW aligned address)**

1B100834 TX_MAX_CNT_3 TX Ring #3 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_MAX_CNT	**The maximum number of TXD count in TX Ring #0**

1B100838 TX_CTX_IDX_3 TX Ring #3 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_CTX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_CTX_IDX	**Point to the next TXD CPU wants to use**

1B10083C TX_DTX_IDX_3 TX Ring #3 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				TX_DTX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	TX_DTX_IDX	**Point to the next TXD DMA wants to use**

1B100900 RX_BASE_PTR_0 RX Ring #0 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #0 (4-DW aligned address)**

1B100904 RX_MAX_CNT_0 RX Ring #0 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

1B100908 RX_CRX_IDX_0 RX Ring #0 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10090C RX_DRX_IDX_0 RX Ring #0 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															

Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B100910 RX_BASE_PTR_1 RX Ring #1 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #0 (4-DW aligned address)**

1B100914 RX_MAX_CNT_1 RX Ring #1 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

1B100918 RX_CRX_IDX_1 RX Ring #1 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10091C RX_DRX_IDX_1 RX Ring #1 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B100920 RX_BASE_PTR_2 RX Ring #2 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #2 (4-DW aligned address)**

1B100924 RX_MAX_CNT_2 RX Ring #2 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**

Bit(s)	Name	Description
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

1B100928 RX_CRX_IDX_2 RX Ring #2 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10092C RX_DRX_IDX_2 RX Ring #2 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B100930 RX_BASE_PTR_3 RX Ring #3 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring #3 (4-DW aligned address)**

1B100934 RX_MAX_CNT_3 RX Ring #3 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #0**

1B100938 RX_CRX_IDX_3 RX Ring #3 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10093C RX_DRX_IDX_3 RX Ring #3 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B100A00 PDMA_INFO PDMA Information 2C000204

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV				INDEX_WIDTH				BASE_PTR_WIDTH							
Type	RO				RO				RO							
Reset	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	RX_RING_NUM								TX_RING_NUM							
Type	RO								RO							
Reset	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0

Bit(s)	Name	Description
31:28	REV	**Reserved**
27:24	INDEX_WIDTH	**Point to the next RXD CPU wants to use**
23:16	BASE_PTR_WIDTH	**Base pointer width, x** Base_addr[31:32-x] is shared with all ring base adderss. Only ring #0 base address[31:32-x] field Is writabl. [note]: "0" means no bit of base_address is shared.
15:8	RX_RING_NUM	**Rx ring number**
7:0	TX_RING_NUM	**Tx ring number**

1B100A04 PDMA_GLO_CFG PDMA Global Configuration 40001450

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_2B_OFFSET	CSR_CLKGATE_BYP	BYTE_SWAP	REV0[15:3]												
Type	RW	RW	RW	DC												
Reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0[2:0]			ADMA_RX_BT_SIZE		MULTI_EN	EXT_FIFO_EN	DESC_32B_E	BIG_ENDIAN	TX_WB_DDONE	PDMA_BT_SIZE		RX_DMA_BUSY	RX_DMA_EN	TX_DMA_BUSY	TX_DMA_EN
Type	DC			RW		RW	RW	RW	RW	RW	RW		RO	RW	RO	RW
Reset	0	0	0	1	0	1	0	0	0	1	0	1	0	0	0	0

Bit(s)	Name	Description
31	RX_2B_OFFSET	**2-Byte offset** 0: Rx-buffer will be 4-byte offset 1: Rx-buffer will be 2-byte offset
30	CSR_CLKGATE_BYP	**Clock gated Bypass** 1: PDMA clock is free-running 0: PDMA clock is gating as idle
29	BYTE_SWAP	**Byte Swap** 0: PDMA will not do byte swapping for TX/RX packet descriptor 1: PDMA will do byte swaping for TX/RX packet descriptor
28:13	REV0	**Reserved**
12:11	ADMA_RX_BT_SIZE	**The burst size of ADMA RX** 0: 4 DWORDs (16-bytes) 1: 8 DWORDs (32-bytes) 2: 16 DWORDs (64-bytes) 3: 32 DWORDs (128-bytes)
10	MULTI_EN	**Enable Multi-burst DMA transfer** 0: disabled 1: Multi-burst DMA enabled
9	EXT_FIFO_EN	**Enable multi-burst DMA transfer for the external FIFO.** 0: Only check internal FIFO 1: Check both internal/external FIFO

Bit(s)	Name	Description
8	DESC_32B_E	**Enable 32-Byte Descriptor Length** 0: PDMA will fetch the next TX/RX descriptor by addign 16-Byte 1: PDMA will fetch the next TX/RX descriptor by addign 32-Byte
7	BIG_ENDIAN	**Big endian** 0: PDMA will not do byte swapping for TX/RX packet header and payload 1: PDMA will do byte swaping for TX/RX packet header and payload
6	TX_WB_DDONE	0: Disable TX_DMA writing back DDONE into TXD 1: Enable TX_DMA writing back DDONE into TXD
5:4	PDMA_BT_SIZE	**The burst size of PDMA** 0: 4 DWORDs (16-bytes) 1: 8 DWORDs (32-bytes) 2: 16 DWORDs (64-bytes) 3: Reserved
3	RX_DMA_BUSY	0: RX_DMA is not busy 1: RX_DMA is busy
2	RX_DMA_EN	0: Diable RX_DMA (when disabled, RX_DMA will finish the current receiving packet, then stop) 1: Enable RX_DMA
1	TX_DMA_BUSY	0: TX_DMA is not busy 1: TX_DMA is busy
0	TX_DMA_EN	0: Disable TX_DMA (when disabled, TX_DMA will finish the current sending packet, then stop) 1: Enable TX_DMA

1B100A08 PDMA_RST_IDX PDMA Reset Index 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0														RST_DRX_IDX1	RST_DRX_IDX0
Type	DC														RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1												RST_DTX_IDX3	RST_DTX_IDX2	RST_DTX_IDX1	RST_DTX_IDX0
Type	DC												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:18	REV0	**Reserved**
17	RST_DRX_IDX1	**Write 1 to reset RX_DMA RX_IDX1 to 0**
16	RST_DRX_IDX0	**Write 1 to reset RX_DMA RX_IDX0 to 0**
15:4	REV1	**Reserved**
3	RST_DTX_IDX3	**Write 1 to reset TX_DMA TX_IDX3 to 0**
2	RST_DTX_IDX2	**Write 1 to reset TX_DMA TX_IDX2 to 0**
1	RST_DTX_IDX1	**Write 1 to reset TX_DMA TX_IDX1 to 0**
0	RST_DTX_IDX0	**Write 1 to reset TX_DMA TX_IDX0 to 0**

1B100A0C DELAY_INT_CFG Delay Interrupt Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TXDLY_INT_EN	TXMAX_PINT							TXMAX_PTIME							
Type	RW	RW							RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RXDLY_INT_EN	RXMAX_PINT							RXMAX_PTIME							
Type	RW	RW							RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TXDLY_INT_EN	**Delay interrupt mechanism** 0: Disable TX delayed interrupt mechanism 1: Enable Tx delayed interrupt mechanism
30:24	TXMAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final TX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
23:16	TXMAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than TXMAX_PTIME x 20us or the number of pended TX_DONE is equal or greater than TXMAX_PINT 9see above), an final TX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.
15	RXDLY_INT_EN	0: Disable Rx delayed interrupt mechanism 1: Enable Rx delayed interrupt mechanism
14:8	RXMAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final RX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
7:0	RXMAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than RXMAX_PTIME x 20us or the number of pended RX_DONE is equal or greater than RXMAX_PINT 9see above), an finalRX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.

1B100A10 FREEQ_THRES Free Queue Threshold 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV1[27:12]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1[11:0]												FREEQ_THRES			
Type	DC												RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	REV1	**Reserved**

Bit(s)	Name	Description
3:0	FREEQ_THRES	**Rx free queue threshold** PDMA will stop DMA interface when left RX descriptors reach this threshold

1B100A20 INT_STATUS Interrupt Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	TX_COHERENT	TX_DLY_INT	RING3_RX_DLY_INT	RING2_RX_DLY_INT	RING1_RX_DLY_INT	RXD_ERROR	ALT_RPLC_INT3	ALT_RPLC_INT2	ALT_RPLC_INT1	REV0	RX_DONE_INT3	RX_DONE_INT2	RX_DONE_INT1	RX_DONE_INT0
Type	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	W1C	DC	W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1												TX_DONE_INT3	TX_DONE_INT2	TX_DONE_INT1	TX_DONE_INT0
Type	DC												W1C	W1C	W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**RX_DMA finds data coherent event while checking ddone bit.**
30	RX_DLY_INT	**Summary of the whole PDMA Rx related interrupts.**
29	TX_COHERENT	**TX_DMA finds data coherent event while checking ddone bit.**
28	TX_DLY_INT	**Summary of the whole PDMA Tx related interrupts.**
27	RING3_RX_DLY_INT	**Rx ring #3 packet receive interrupt**
26	RING2_RX_DLY_INT	**Rx ring #2 packet receive interrupt**
25	RING1_RX_DLY_INT	**Rx ring #1 packet receive interrupt**
24	RXD_ERROR	**Rx descriptor SDL0 error interrupt**
23	ALT_RPLC_INT3	**Auto-learn otg3 replacement interrupt**
22	ALT_RPLC_INT2	**Auto-learn otg2 replacement interrupt**
21	ALT_RPLC_INT1	**Auto-learn otg1 replacement interrupt**
20	REV0	**Reserved**
19	RX_DONE_INT3	**Rx ring #3 packet receive interrupt**
18	RX_DONE_INT2	**Rx ring #2 packet receive interrupt**
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt**
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt**
15:4	REV1	**Reserved**
3	TX_DONE_INT3	**Tx ring #3 packet transmit interrupt**
2	TX_DONE_INT2	**Tx ring #2 packet transmit interrupt**
1	TX_DONE_INT1	**Tx ring #1 packet transmit interrupt**
0	TX_DONE_INT0	**Tx ring #0 packet transmit interrupt**

1B100A28 INT_MASK Interrupt Mask 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	TX_COHERENT	TX_DLY_INT	REV0										RX_DONE_INT1	RX_DONE_INT0

Type	RW	RW	RW	W1C	DC										RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1												TX_DONE_INT3	TX_DONE_INT2	TX_DONE_INT1	TX_DONE_INT0
Type	DC												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**Interrupt enable for RX_DMA data coherent vent** 0: Disable interrupt 1: Enable interrupt
30	RX_DLY_INT	**Summary of the whole PDMA Rx related interrupts.** 0: Disable interrupt 1: Enable interrupt
29	TX_COHERENT	**Interrupt enable for TX_DMA data coherent vent** 0: Disable interrupt 1: Enable interrupt
28	TX_DLY_INT	**Summary of the whole PDMA Tx related interrupts.** 0: Disable interrupt 1: Enable interrupt
27:18	REV0	**Reserved** 0: Disable interrupt 1: Enable interrupt
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
15:4	REV1	**Reserved** 0: Disable interrupt 1: Enable interrupt
3	TX_DONE_INT3	**Tx ring #3 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
2	TX_DONE_INT2	**Tx ring #2 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
1	TX_DONE_INT1	**Tx ring #1 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt
0	TX_DONE_INT0	**Tx ring #0 packet transmit interrupt** 0: Disable interrupt 1: Enable interrupt

1B100A40 PDMA_INT1_VEC_GRP0 PDMA Interrupt Status Group 0 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDMA_INT1_VEC_GRP0[31:16]															
Type	RO															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDMA_INT1_VEC_GRP0[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PDMA_INT1_VEC_GRP0	**Interrupt group 0 status, this information already "and" with "~PDMA_INT_GRP1 & ~PDMA_INT_GRP2"** Each bit definition is same as "INT_STATUS"

1B100A44 PDMA_INT1_VEC_GRP1 PDMA Interrupt Status Group 1 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDMA_INT1_VEC_GRP1[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDMA_INT1_VEC_GRP1[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PDMA_INT1_VEC_GRP1	**Interrupt group 1 status, this information already "and" with "PDMA_INT_GRP1"** Each bit definition is same as "INT_STATUS"

1B100A48 PDMA_INT1_VEC_GRP2 PDMA Interrupt Status Group 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDMA_INT1_VEC_GRP2[23:8]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDMA_INT1_VEC_GRP2[7:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0								

Bit(s)	Name	Description
31:8	PDMA_INT1_VEC_GRP2	**Interrupt group 2 status, this information already "and" with "PDMA_INT_GRP2"** Each bit definition is same as "INT_STATUS"

1B100A50 PDMA_INT_GRP1 PDMA Interrupt Group 1 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDMA_INT_GRP1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	PDMA_INT_GRP1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PDMA_INT_GRP1	**Interrupt group 1 assignment.** Each bit's definition is same as "INT_STATUS" 0: Leave to ADMA int group 0, if (PDMA_INT_GRP1[n]==0 & PDMA_INT_GRP2[n]==0) 1: Assign to PDMA int group 1

1B100A54 PDMA_INT_GRP2 PDMA Interrupt Group 2 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PDMA_INT_GRP2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	PDMA_INT_GRP2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	PDMA_INT_GRP2	**Interrupt group 2 assignment.** Each bit's definition is same as "INT_STATUS" 0: Leave to ADMA int group 0, if (PDMA_INT_GRP1[n]==0 & PDMA_INT_GRP2[n]==0) 1: Assign to PDMA int group 2

1B101400 CDMQ_IG_CTRL CDM VLAN Control 81000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	CDM_TPID															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0														UNTAG_EN	STAG_EN
Type	DC														RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	CDM_TPID	**Inserted VLAN TPID**
15:2	REV0	**Reserved**
1	UNTAG_EN	**VLAN untagging** Untagg the egress packets which are transmitted from CDM 0: disable 1: The first 4-bytes VLAN tag after Source Address will be untagged
0	STAG_EN	**Special tag indication** Indicate that the received packets is carrying the special tag from CPU . 0: no special tag inserted. 1: The first 2-byte after Source Address is the special tag.

1B101404 CDMQ_EG_CTRL CDM Egress Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0[30:15]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0[14:0]															UNTAG_EN
Type	DC															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:1	REV0	**Reserved**
0	UNTAG_EN	**VLAN untag** Untag the egress packets which are transmitted from CDM to CPU 0: disable 1: The first 4-bytes VLAN tag after Source Address will be untagged

1B101408 CDMQ_PPP_GEN CDM PPPoE Generation 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0															PPP_INS
Type	DC															RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SESS_ID															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:17	REV0	**Reserved**
16	PPP_INS	**PPPoE Header Insertion**
15:0	SESS_ID	**PPPoE Session ID**

1B101500 GDM2_IG_CTRL GDM Ingress Control 00717777

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0						INSV_EN	STAG_EN	REV1	GDM_ICS_EN	GDM_TCS_EN	GDM_UCS_EN	DROP_256B	REV2		STRP_CRC
Type	DC						RW	RW	DC	RW	RW	RW	RW	DC		RW
Reset	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MYMAC_DP				BC_DP				MC_DP				UN_DP			
Type	RW				RW				RW				RW			
Reset	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1

Bit(s)	Name	Description
31:26	REV0	**Reserved**
25	INSV_EN	**VLAN insertion** Insert Port VID on the received packets on the corresponding GDM port. 0: disable 1: Insert 4-byte VLAN tag after Source Address
24	STAG_EN	**Special tag indication** Indicate that the received packets is carrying the special tag from GDM port. 0: no special tag inserted. 1: The first 2-byte after Source Address is the special tag.
23	REV1	**Reserved**
22	GDM_ICS_EN	**IPv4 header checksum error drop** 0: checksum error status reported on RX descriptor (IP4F) 1: checksum error packet will be dropped
21	GDM_TCS_EN	**TCP checksum error drop** 0: checksum error status reported on RX descriptor (L4F) 1: checksum error packet will be dropped
20	GDM_UCS_EN	**UDP checksum error drop** 0: checksum error status reported on RX descriptor (L4F) 1: checksum error packet will be dropped
19	DROP_256B	**A special mode to drop packets with payload > 256-bytes** 1'b0: Drop packets according to the standard Ethernet frame length limitation. 1'b1: Dopr packets with payloa > 256 bytes
18:17	REV2	**Reserved**
16	STRP_CRC	**GDM RX CRC Stripping** 1'b: Disable GDM RX CRC stripping 1'b1: Enable GDM RX CRC stripping
15:12	MYMAC_DP	**MY_MAC frames destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
11:8	BC_DP	**Broadcast frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved
7:4	MC_DP	**Multicast frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard

Bit(s)	Name	Description
		4'b1xxx: Reserved
3:0	UN_DP	**Other frame destination port** 4'b0000: PDMA 4'b0001: GDM1 4'b0010: GDM2 4'b0011: Reserved 4'b0100: PPE 4'b0101: QDMA 4'b0110: Reserved 4'b0111: Discard 4'b1xxx: Reserved

1B101504 GDM2_EG_CTRL GDM Egress Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0	UNTAG_EN	DIS_PAD	DIS_CRC	REV1			SHPR_EN	BK_SIZE							
Type	DC	RW	RW	RW	DC			RW	RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TK_TICK	REV2	TK_RATE													
Type	RW	DC	RW													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	REV0	
30	UNTAG_EN	**VLAN untagging** Untagg the egress packets which are transmitted from GDM. 0: disable 1: The first 4-bytes VLAN tag after Source Address will be untagged
29	DIS_PAD	**GMAC Tx padding fucntion** 1'b0: Enable GMAC Tx padding 1'b1: Disable GMAC Tx padding
28	DIS_CRC	**GMAC TX CRC generation** 1'b0: Enable GMAC Tx CRC generation 1'b1: Disable GMAC Tx CRC generation
27:25	REV1	**Reserved**
24	SHPR_EN	**GDM output shaper enable** 1'b0: Disable 1'b1: Enable
23:16	BK_SIZE	**GDM output shaper bucket size.** This unit is 1K-byte
15	TK_TICK	**GDM shaper token period** 1'b0: GDM shaper add token every 1ms 1'b1: GDM shaper add token every 20us
14	REV2	**Reserved**
13:0	TK_RATE	**GDM output shaper token rate** The unit is 8-Byte/ms or 8-Byte/20us according to TK_TICK.

1B101508 GDM2_MAC_LSB GDM MY_MAC Address LSB 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAC_ADR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAC_ADR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAC_ADR	**MY_MAC Address [31:0]**

1B10150C GDM2_MAC_MSB GDM MY_MAC Address MSB 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAC_ADR															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	RESV0	**Reserved**
15:0	MAC_ADR	**MY_MAC Address [47:32]**

1B101510 GDM2_VLAN_GEN GDM VLAN Generation 81000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM_TPID															
Type	RW															
Reset	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GDM_PRI			GDM_CFI	GDM_VID											
Type	RW			RW	RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	GDM_TPID	**Inserted VLAN TPID**
15:13	GDM_PRI	**Inserted PRI**
12	GDM_CFI	**Inserted CFI**
11:0	GDM_VID	**Inserted VLAN ID**

1B101514 GDM2_FILTER_CRTL GDM Filter Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0[27:12]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV0[11:0]												GDM_VIDF_EN	GMD_HASH_ALG	GDM_DAF_MODE	
Type	DC												RW	RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:4	REV0	
3	GDM_VIDF_EN	**GDM VLAN ID filter enable**
2	GMD_HASH_ALG	**Hash Algorithm setting** 0: Direct map - Using DA40 and DA 7~0 as hash key. Receive packet if corresponding bit is set. 1: CRC32 - Using 32 bit CRC bit 8~0 of DA as hash key. Receive packet if corresponding bit is set.
1:0	GDM_DAF_MODE	**Receive packet with DA Filter function** 0: Promisculous mode - Receive all packets. 1: Filter packet by MAC_MAC/Broadcast/Hash table 2: Filter packet by MAC_MAC/Broadcast 3: Reserved

1B101518 GDM2_VIDF01 GDM VID Filter Control 01 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM2_VID1_VLD	REV1			GDM2_VID1											
Type	RW	DC			RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GDM2_VID0_VLD	REV0			GDM2_VID0											
Type	RW	DC			RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	GDM2_VID1_VLD	**VLAN ID #1 valid bit**
30:28	REV1	
27:16	GDM2_VID1	**VLAN ID #1**
15	GDM2_VID0_VLD	**VLAN ID #0 valid bit**
14:12	REV0	
11:0	GDM2_VID0	**VLAN ID #0**

1B10151C GDM2_VIDF23 GDM VID Filter Control 23 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	GDM2_VID3_VLD	REV1			GDM2_VID3											
Type	RW	DC			RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	GDM2_VID2_VLD	REV0			GDM2_VID2											
Type	RW	DC			RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	GDM2_VID3_VLD	**VLAN ID #3 valid bit**
30:28	REV1	
27:16	GDM2_VID3	**VLAN ID #3**
15	GDM2_VID2_VLD	**VLAN ID #2 valid bit**
14:12	REV0	
11:0	GDM2_VID2	**VLAN ID #2**

1B101800 QTX_CFG_0 TX Queue #0 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES.
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES.

1B101804 QTX_SCH_0 TX Queue #0 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Leaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** [Note] Rate control is calculated by 125MHz bus clock, please multiply by a factor when the bus clock is not 125MHz. 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable.** [Note] Rate control is calculated by 125MHz bus clock, please multiply by a factor when the bus clock is not 125MHz. 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101808 QTX_HEAD_0 TX Queue #0 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10180C QTX_TAIL_0 TX Queue #0 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101810 QTX_CFG_1 TX Queue #1 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES.
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES.

1B101814 QTX_SCH_1 TX Queue #1 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** [Note] Rate control is calculated by 125MHz bus clock, please multiply by a factor when the bus clock is not 125MHz. 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable.** [Note] Rate control is calculated by 125MHz bus clock, please multiply by a factor when the bus clock is not 125MHz. 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101818 QTX_HEAD_1 TX Queue #1 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10181C QTX_TAIL_1 TX Queue #1 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101820 QTX_CFG_2 TX Queue #2 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES.
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES.

1B101824 QTX_SCH_2 TX Queue #2 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_S	LEAKY_BK	BUCKET_DEP		MIN_RATE_E	MIN_RATE_MAN							MIN_RATE_EXP			

	EL				N											
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101828 QTX_HEAD_2 TX Queue #2 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10182C QTX_TAIL_2 TX Queue #2 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101830 QTX_CFG_3 TX Queue #3 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101834 QTX_SCH_3 TX Queue #3 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps)

5: QDMA_RATE_EXP5 (default: 100Mbps)
Others: 20'd1, 1Gbps

1B101838 QTX_HEAD_3 TX Queue #3 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10183C QTX_TAIL_3 TX Queue #3 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101840 QTX_CFG_4 TX Queue #4 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer

1: Buffer Reserved

1B101844 QTX_SCH_4 TX Queue #4 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps)

1: QDMA_RATE_EXP1 (default: 10kbps)
2: QDMA_RATE_EXP2 (default: 100kbps)
3: QDMA_RATE_EXP3 (default: 1Mbps)
4: QDMA_RATE_EXP4 (default: 10Mbps)
5: QDMA_RATE_EXP5 (default: 100Mbps)
Others: 20'd1, 1Gbps

1B101848 QTX_HEAD_4 TX Queue #4 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10184C QTX_TAIL_4 TX Queue #4 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101850 QTX_CFG_5 TX Queue #5 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101854 QTX_SCH_5 TX Queue #5 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127

3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101858 QTX_HEAD_5 TX Queue #5 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10185C QTX_TAIL_5 TX Queue #5 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101860 QTX_CFG_6 TX Queue #6 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path**

		The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101864 QTX_SCH_6 TX Queue #6 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate)

		1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101868 QTX_HEAD_6 TX Queue #6 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10186C QTX_TAIL_6 TX Queue #6 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101870 QTX_CFG_7 TX Queue #7 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101874 QTX_SCH_7 TX Queue #7 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n

		15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101878 QTX_HEAD_7 TX Queue #7 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10187C QTX_TAIL_7 TX Queue #7 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101880 QTX_CFG_8 TX Queue #8 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							

Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101884 QTX_SCH_8 TX Queue #8 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.**

		0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101888 QTX_HEAD_8 TX Queue #8 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10188C QTX_TAIL_8 TX Queue #8 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101890 QTX_CFG_9 TX Queue #9 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B101894 QTX_SCH_9 TX Queue #9 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps)

		5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101898 QTX_HEAD_9 TX Queue #9 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B10189C QTX_TAIL_9 TX Queue #9 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018A0 QTX_CFG_10 TX Queue #10 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018A4 QTX_SCH_10 TX Queue #10 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps)

		1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018A8 QTX_HEAD_10 TX Queue #10 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018AC QTX_TAIL_10 TX Queue #10 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018B0 QTX_CFG_11 TX Queue #11 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018B4 QTX_SCH_11 TX Queue #11 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127

19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018B8 QTX_HEAD_11 TX Queue #11 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018BC QTX_TAIL_11 TX Queue #11 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018C0 QTX_CFG_12 TX Queue #12 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018C4 QTX_SCH_12 TX Queue #12 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper

26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018C8 QTX_HEAD_12 TX Queue #12 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018CC QTX_TAIL_12 TX Queue #12 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018D0 QTX_CFG_13 TX Queue #13 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018D4 QTX_SCH_13 TX Queue #13 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB

27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018D8 QTX_HEAD_13 TX Queue #13 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018DC QTX_TAIL_13 TX Queue #13 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															

Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018E0 QTX_CFG_14 TX Queue #14 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018E4 QTX_SCH_14 TX Queue #14 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler 1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth**

		0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable max. shaper
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018E8 QTX_HEAD_14 TX Queue #14 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018EC QTX_TAIL_14 TX Queue #14 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B1018F0 QTX_CFG_15 TX Queue #15 Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_CNT															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_RESV_CNT								SW_RESV_CNT							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	TX_DES_CNT	**TX descriptor count in Queue #**
15:8	HW_RESV_CNT	**Buffer Reserved for HW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved
7:0	SW_RESV_CNT	**Buffer Reserved for SW path** The reserved buffer number is specified on QDMA_RES_THRES. 0: No reserved buffer 1: Buffer Reserved

1B1018F4 QTX_SCH_15 TX Queue #15 Schedule 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_SCH_SEL	LEAKY_BK	BUCKET_DEP		MIN_RATE_EN	MIN_RATE_MAN							MIN_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_RATE_WGHT				MAX_RATE_EN	MAX_RATE_MAN							MAX_RATE_EXP			
Type	RW				RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	TX_SCH_SEL	**TX queue on scheduler no.** 0: SCH1 scheduler

		1: SCH2 scheduler
30	LEAKY_BK	**Eleaky Bucket Select**
29:28	BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	MIN_RATE_EN	**TX Queue min. rate control enable** 0: disable. When disable, shaper will always let the pakcet pass(infinite rate) 1: enable min. shaper
26:20	MIN_RATE_MAN	**Mantissa part of the min. rate control of the TX queue #** value rang: 0 .. 127
19:16	MIN_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15:12	MAX_RATE_WGHT	**The weighted value of the WFQ for TX queue # maximum rate.** 0: Weight value = 16 1: Weight value = 1 2: Weight value = 2 n: Weight value = n 15: Weight value = 15
11	MAX_RATE_EN	**TX Queue max. rate control enable**
10:4	MAX_RATE_MAN	**Mantissa part of the max. rate control of the TX queue #** value rang: 0 .. 127
3:0	MAX_RATE_EXP	**Exponent part of the max. rate control of the TX queue #** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B1018F8 QTX_HEAD_15 TX Queue #15 Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_HEAD[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_HEAD[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_HEAD	**TX descriptor head pointer in Queue #**

1B1018FC QTX_TAIL_15 TX Queue #15 Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	TX_DES_TAIL[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	TX_DES_TAIL[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	TX_DES_TAIL	**TX descriptor tail pointer in Queue #**

1B101900 QRX_BASE_PTR_0 RX Ring #0 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring # (4-DW aligned address)**

1B101904 QRX_MAX_CNT_0 RX Ring #0 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #**

1B101908 QRX_CRX_IDX_0 RX Ring #0 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															

Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10190C QRX_DRX_IDX0 RX Ring #0 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B101910 QRX_BASE_PTR_1 RX Ring #1 Base Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_BASE_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_BASE_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RX_BASE_PTR	**Point to the base address of RX Ring # (4-DW aligned address)**

1B101914 QRX_MAX_CNT_1 RX Ring #1 Maximum Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_MAX_CNT											

Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_MAX_CNT	**The maximum number of RXD count in RX Ring #**

1B101918 QRX_CRX_IDX1 RX Ring #1 CPU pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_CRX_IDX											
Type	DC				RW											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_CRX_IDX	**Point to the next RXD CPU wants to use**

1B10191C QRX_DRX_IDX1 RX Ring #1 DMA poitner 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV[19:4]															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV[3:0]				RX_DRX_IDX											
Type	DC				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:12	REV	**Reserved**
11:0	RX_DRX_IDX	**Point to the next RXD DMA wants to use**

1B101A00 QDMA_INFO QDMA Information 000C0210

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV				INDEX_WIDTH											
Type	RO				RO											
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RX_RING_NUM								TX_QUE_NUM							
Type	RO								RO							
Reset	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0

Bit(s)	Name	Description

31:28	REV	**QDMA revision**
27:16	INDEX_WIDTH	**Ring index width**
15:8	RX_RING_NUM	**Rx ring number**
7:0	TX_QUE_NUM	**Tx queue number**

1B101A04 QDMA_GLO_CFG QDMA Global Configuration 40000010

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_2B_OFFSET	CSR_CLKGATE	BYTE_SWAP										RBUS_CO_PRO			
Type	RW	RW	RW										RW			
Reset	0	1	0										0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name						NDP_CO_PRO	BKTCLK_CTRL	DESC_32B_E	BIG_ENDIAN	TX_WB_DDONE	QDMA_BT_SIZE		RX_DMA_BUSY	RX_DMA_EN	TX_DMA_BUSY	TX_DMA_EN
Type						RW	RW	RW	RW	RW	RW		RO	RW	RO	RW
Reset						0	0	0	0	0	0	1	0	0	0	0

Bit(s)	Name	Description
31	RX_2B_OFFSET	**2-Byte offset** 0: Rx-buffer will be 4-byte offset 1: Rx-buffer will be 2-byte offset
30	CSR_CLKGATE	**Clock gated** 0: QDMA clock is free-running 1: QDMA clock is gating as idle
29	BYTE_SWAP	**Byte Swap** 0: QDMA will not do byte swapping for TX/RX packet descriptor 1: QDMA will do byte swaping for TX/RX packet descriptor
19:16	RBUS_CO_PRO	**Rbus Coherence Protection** 4'b0000: Disable Rbus Coherence Protection, read request of QDMA rbus interface will be assertted back to back. 4'b1000: Read request will be postponed till all past write complete. 4'b1001: Read request will be postponed till all past write complete + extra 1*16 system clock cycles. 4'b1010: Read request will be postponed till all past write complete + extra 2*16 system clock cycles. 4'b1111: Read request will be postponed till all past write complete + extra 7*16 system clock cycles.
10	NDP_CO_PRO	**Next Descriptor Pointer Coherence Protection** 0: Disable the NDP coherence protection 1: Enable the NDP coherence protection. Before de-queue a packet, QDMA TX will check that descriptor's NDP[0] already be set as 1 by Forwarding agent.
9	BKTCLK_CTRL	**Scheduler bucket clock control** 0: Scheduler bucket clock is 125Mhz 1: Scheduler bucket clock is 62.5Mhz
8	DESC_32B_E	**Enable 32-Byte RX Descriptor Length** 0: PDMA will fetch the next RX descriptor by addign 16-Byte 1: PDMA will fetch the next RX descriptor by addign 32-Byte
7	BIG_ENDIAN	**Big endian** 0: QDMA will not do byte swapping for TX/RX packet header and payload

		1: QDMA will do byte swaping for TX/RX packet header and payload
6	TX_WB_DDONE	0: Disable TX_DMA writing back DDONE into TXD 1: Enable TX_DMA writing back DDONE into TXD
5:4	QDMA_BT_SIZE	**The burst size of QDMA** 0: 4 DWORDs (16-bytes) 1: 8 DWORDs (32-bytes) 2: 16 DWORDs (64-bytes) 3: Reserved
3	RX_DMA_BUSY	0: RX_DMA is not busy 1: RX_DMA is busy
2	RX_DMA_EN	0: Diable RX_DMA (when disabled, RX_DMA will finish the current receiving packet, then stop) 1: Enable RX_DMA
1	TX_DMA_BUSY	0: TX_DMA is not busy 1: TX_DNA is busy
0	TX_DMA_EN	0: Disable TX_DMA (when disabled, TX_DMA will finish the current sending packet, then stop) 1: Enable TX_DMA

1B101A08 QDMA_RST_IDX QDMA Reset Index 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0														RST_DRX_IDX1	RST_DRX_IDX0
Type	DC														RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RST_TX_IDX15	RST_TX_IDX14	RST_TX_IDX13	RST_TX_IDX12	RST_TX_IDX11	RST_TX_IDX10	RST_TX_IDX9	RST_TX_IDX8	RST_TX_IDX7	RST_TX_IDX6	RST_TX_IDX5	RST_TX_IDX4	RST_TX_IDX3	RST_TX_IDX2	RST_TX_IDX1	RST_TX_IDX0
Type	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:18	REV0	**Reserved**
17	RST_DRX_IDX1	**Write 1 to reset RX_DMA RX_IDX1 to 0**
16	RST_DRX_IDX0	**Write 1 to reset RX_DMA RX_IDX0 to 0**
15	RST_TX_IDX15	**Write 1 to reset TX_DMA TX_IDX15 to 0**
14	RST_TX_IDX14	**Write 1 to reset TX_DMA TX_IDX14 to 0**
13	RST_TX_IDX13	**Write 1 to reset TX_DMA TX_IDX13 to 0**
12	RST_TX_IDX12	**Write 1 to reset TX_DMA TX_IDX12 to 0**
11	RST_TX_IDX11	**Write 1 to reset TX_DMA TX_IDX11 to 0**
10	RST_TX_IDX10	**Write 1 to reset TX_DMA TX_IDX10 to 0**
9	RST_TX_IDX9	**Write 1 to reset TX_DMA TX_IDX9 to 0**
8	RST_TX_IDX8	**Write 1 to reset TX_DMA TX_IDX8 to 0**
7	RST_TX_IDX7	**Write 1 to reset TX_DMA TX_IDX7 to 0**
6	RST_TX_IDX6	**Write 1 to reset TX_DMA TX_IDX6 to 0**
5	RST_TX_IDX5	**Write 1 to reset TX_DMA TX_IDX5 to 0**
4	RST_TX_IDX4	**Write 1 to reset TX_DMA TX_IDX4 to 0**

3	RST_TX_IDX3	**Write 1 to reset TX_DMA TX_IDX3 to 0**
2	RST_TX_IDX2	**Write 1 to reset TX_DMA TX_IDX2 to 0**
1	RST_TX_IDX1	**Write 1 to reset TX_DMA TX_IDX1 to 0**
0	RST_TX_IDX0	**Write 1 to reset TX_DMA TX_IDX0 to 0**

1B101A0C QDMA_DELAY_INT Delay Interrupt Configuration 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RXDLY_INT_EN	RXMAX_PINT							RXMAX_PTIME							
Type	RW	RW							RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RLS_DINT_EN	RLS_MAX_PINT							RLS_MAX_PTIME							
Type	RW	RW							RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RXDLY_INT_EN	0: Disable Rx delayed interrupt mechanism 1: Enable Rx delayed interrupt mechanism
30:24	RXMAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final RX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
23:16	RXMAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than RXMAX_PTIME x 20us or the number of pended RX_DONE is equal or greater than RXMAX_PINT 9see above), an finalRX_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.
15	RLS_DINT_EN	0: Disable RLS delayed interrupt mechanism 1: Enable RLS delayed interrupt mechanism
14:8	RLS_MAX_PINT	**Specified Max. number of pended interrupts** When the number of pended interrupts is equal or greater than the value specified here or interupt pending time reach the limit (see below), an final RLS_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt count check.
7:0	RLS_MAX_PTIME	**Specified Max. pended time** When the pending time is equal or greater than RLS_MAX_PTIME x 20us or the number of pended RX_DONE is equal or greater than RLS_MAX_PINT (see above), an final RLS_DLY_INT is generated. [Note] reset to 0 can disable pending interrupt time check.

1B101A10 QDMA_FC_THRES Flow ControlThreshold 90904444

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SW_DROP_FSTVQ_MODE		SW_DROP_MODE		SW_DROP		SW_DROP	SW_DROP	HW_DROP_FSTVQ_MODE		HW_DROP_MODE		HW_DROP		HW_DROP	HW_DROP

					FSTVQ		FFA	EN					FSTVQ		FFA	EN
Type	RW		RW		RW		RW	RW	RW		RW		RW		RW	RW
Reset	1	0	0	1	0		0	0	1	0	0	1	0		0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	SHARE_SW_TH				SHARE_HW_TH				FREE_TH				RING_TH			
Type	RW				RW				RW				RW			
Reset	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0

Bit(s)	Name	Description
31:30	SW_DROP_FSTVQ_MODE	**SW drop condition on fullest virtual queue** If the corresponding PQ SW occupied queue depth over the SW_RESV_CNT, the enqueuing packet to fullest VQ will be drop depends on the usage of SW free-for-all buffer. 2'b00: FFA usage >= 0% 2'b01: FFA usage >= 25% 2'b10: FFA usage >= 50% 2'b11: FFA usage >= 75%
29:28	SW_DROP_MODE	**Drop Distribution Mode for SW Drop** 2'b00: 50/25/0/0 2'b01: 75/50/25/0 2'b10: 100/75/50/25 2'b11: 100/100/75/50
27	SW_DROP_FSTVQ	**SW drop enqueuing packet on fullest virtual queue**
25	SW_DROP_FFA	**Flow Control Drop** QDMA asserts FC when FFA is equal to 0 and egress queue count is bigger than the reserved queue count. However, the de-assert condition is specified on this control bit. 0: QDMA deassert FC when the egress queue count is less than the reserved queu count. 1: QDMA deassert FC when FFA is bigger than 0 or the egress queue count is less than the reserved queue count.
24	SW_DROP_EN	**Enable Flow Control for SW buffer pool** QDMA drop packets when FFA is equal to 0 and the egress queue count is bigger than the reserved queue count.
23:22	HW_DROP_FSTVQ_MODE	**HW drop condition on fullest virtual queue** If the corresponding PQ SW occupied queue depth over the SW_RESV_CNT, the enqueuing packet to fullest VQ will be drop depends on the usage of HW free-for-all buffer. 2'b00: FFA usage >= 0% 2'b01: FFA usage >= 25% 2'b10: FFA usage >= 50% 2'b11: FFA usage >= 75%
21:20	HW_DROP_MODE	**Drop Distribution Mode for HW Drop** 2'b00: 50/25/0/0 2'b01: 75/50/25/0 2'b10: 100/75/50/25 2'b11: 100/100/75/50
19	HW_DROP_FSTVQ	**HW drop enqueuing packet on fullest virtual queue**
17	HW_DROP_FFA	**Flow Control Drop** QDMA asserts FC when FFA is equal to 0 and the egress queue count is bigger than the reserved queue count. However, the de-assert condition is specified on this control bit. 0: QDMA deassert FC when the egress queue count is less than the reserved queu count. 1: QDMA deassert FC when FFA is bigger than 0 or the egress queue count is less than the reserved queue count.

16	HW_DROP_EN	**Enable Flow Control for HW buffer pool** QDMA drop packets when FFA is equal to 0 and the egress queue count is bigger than the reserved queue count.
15:12	SHARE_SW_TH	**SW Shared buffer threshold** QDMA will drop TX packets when the left shared buffer descriptors reach this threshold
11:8	SHARE_HW_TH	**HW Shared buffer threshold** QDMA will drop TX packets when the left shared buffer descriptors reach this threshold
7:4	FREE_TH	**Rx free buffer threshold** QDMA will pause RXDMA interface when the left free buffer descriptors reach this threshold
3:0	RING_TH	**Rx Ring threshold** QDMA will pause RXDMA interface when the left RX ring descriptors reach this threshold

1B101A14 QDMA_TX_SCH TX Scheduler Rate Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0	SCH2_LEAKY_BK	SCH2_BUCKET_DEP		SCH2_RATE_EN	SCH2_RATE_MAN							SCH2_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1	SCH1_LEAKY_BK	SCH1_BUCKET_DEP		SCH1_RATE_EN	SCH1_RATE_MAN							SCH1_RATE_EXP			
Type	RW	RW	RW		RW	RW							RW			
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	REV0	**Reserved**
30	SCH2_LEAKY_BK	**Leaky Bucket Select**
29:28	SCH2_BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
27	SCH2_RATE_EN	**TX SCH2 max. rate control enable**
26:20	SCH2_RATE_MAN	**Mantissa part of the max. rate control of the TX SCH2** value rang: 0 .. 127
19:16	SCH2_RATE_EXP	**Exponent part of the max. rate control of the TX SCH2** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps
15	REV1	**Reserved**

14	SCH1_LEAKY_BK	**Leaky Bucket Select**
13:12	SCH1_BUCKET_DEP	**Bucket Depth** 0: 2KB 1: 4KB 2: 8KB 3: 16KB
11	SCH1_RATE_EN	**TX SCH1 max. rate control enable**
10:4	SCH1_RATE_MAN	**Mantissa part of the max. rate control of the TX SCH1** value rang: 0 .. 127
3:0	SCH1_RATE_EXP	**Exponent part of the max. rate control of the TX SCH2** value selection 0: QDMA_RATE_EXP0 (default: 1kbps) 1: QDMA_RATE_EXP1 (default: 10kbps) 2: QDMA_RATE_EXP2 (default: 100kbps) 3: QDMA_RATE_EXP3 (default: 1Mbps) 4: QDMA_RATE_EXP4 (default: 10Mbps) 5: QDMA_RATE_EXP5 (default: 100Mbps) Others: 20'd1, 1Gbps

1B101A18 QDMA_INT_STS Interrupt Status **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	RLS_COHERENT	RLS_DLY_INT	REV3										RX_DONE_INT1	RX_DONE_INT0
Type	W1C	W1C	W1C	W1C	DC										W1C	W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VQDEP_DEQ_ERR1	VQDEP_DEQ_ERR0	VQDEP_ENQ_ERR1	VQDEP_ENQ_ERR0	REV2			FWD_FIFO_OV	REV1			QTX_REQ_EMP_VQ	REV0			RLS_DONE_INT
Type	W1C	W1C	W1C	W1C	DC			W1C	DC			W1C	DC			W1C
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**RX_DMA finds data coherent event while checking ddone bit.**
30	RX_DLY_INT	**Summary of the whole QDMA Rx related interrupts.**
29	RLS_COHERENT	**TX_DMA finds data coherent event while checking ddone bit.**
28	RLS_DLY_INT	**Summary of the whole QDMA Tx related interrupts.**
27:18	REV3	**Reserved**
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt**
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt**
15	VQDEP_DEQ_ERR1	**VQ depth mismatch with the VQHPTR & VQTPTR during de-queuing.** A VQ not empty yet after de-queuing, but the remainder vq_depth>=1
14	VQDEP_DEQ_ERR0	**VQ depth mismatch with the VQHPTR & VQTPTR during de-queuing.** A VQ becomes empty after de-queuing but the remainder vq_depth=0
13	VQDEP_ENQ_ERR1	**VQ depth mismatch with the VQHPTR & VQTPTR during en-queuing.** A VQ be en-queued the first packet descirptor, but the vq_depth already >=1.
12	VQDEP_ENQ_ERR0	**VQ depth mismatch with the VQHPTR & VQTPTR during en-queuing.** A VQ be en-queued a packet descirptor, but the vq_depth already full = 16'hffff.
11:9	REV2	**Reserved**

8	FWD_FIFO_OV	**Forwarding engine FIFO overflow interrupt**
7:5	REV1	**Reserved**
4	QTX_REQ_EMP_VQ	**QDMA TX request to transmit an empty VQ**
3:1	REV0	**Reserved**
0	RLS_DONE_INT	**CPU enqueue transmitted packet interrupt**

1B101A1C **QDMA_INT_MASK** **Interrupt Mask** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RX_COHERENT	RX_DLY_INT	RLS_COHERENT	RLS_DLY_INT	REV3										RX_DONE_INT1	RX_DONE_INT0
Type	RW	RW	RW	W1C	DC										RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	VQDEP_DEQ_ERR1	VQDEP_DEQ_ERR0	VQDEP_ENQ_ERR1	VQDEP_ENQ_ERR0	REV2			FWD_FIFO_OV	REV1			QTX_REQ_EMP_VQ	REV0			RLS_DONE_INT
Type	RW	RW	RW	RW	DC			RW	DC			W1C	DC			RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	RX_COHERENT	**Interrupt enable for RX_DMA data coherent vent** 0: Disable interrupt 1: Enable interrupt
30	RX_DLY_INT	**Summary of the whole QDMA Rx related interrupts.** 0: Disable interrupt 1: Enable interrupt
29	RLS_COHERENT	**Interrupt enable for TX_DMA data coherent vent** 0: Disable interrupt 1: Enable interrupt
28	RLS_DLY_INT	**Summary of the whole QDMA Tx related interrupts.** 0: Disable interrupt 1: Enable interrupt
27:18	REV3	**Reserved** 0: Disable interrupt 1: Enable interrupt
17	RX_DONE_INT1	**Rx ring #1 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
16	RX_DONE_INT0	**Rx ring #0 packet receive interrupt** 0: Disable interrupt 1: Enable interrupt
15	VQDEP_DEQ_ERR1	**VQ depth mismatch with the VQHPTR & VQTPTR during de-queuing.** A VQ not empty yet after de-queuing, but the remainder vq_depth>=1 0: Disable interrupt 1: Enable interrupt
14	VQDEP_DEQ_ERR0	**VQ depth mismatch with the VQHPTR & VQTPTR during de-queuing.** A VQ becomes empty after de-queuing but the remainder vq_depth=0 0: Disable interrupt 1: Enable interrupt

13	VQDEP_ENQ_ERR1	**VQ depth mismatch with the VQHPTR & VQTPTR during en-queuing.** A VQ be en-queued the first packet descirptor, but the vq_depth already >=1. 0: Disable interrupt 1: Enable interrupt
12	VQDEP_ENQ_ERR0	**VQ depth mismatch with the VQHPTR & VQTPTR during en-queuing.** A VQ be en-queued a packet descirptor, but the vq_depth already full = 16'hffff. 0: Disable interrupt 1: Enable interrupt
11:9	REV2	**Reserved**
8	FWD_FIFO_OV	**Forwarding engine FIFO overflow interrupt** 0: Disable interrupt 1: Enable interrupt
7:5	REV1	**Reserved**
4	QTX_REQ_EMP_VQ	**QDMA TX request to transmit an empty VQ**
3:1	REV0	**Reserved**
0	RLS_DONE_INT	**CPU release transmitted packet interrupt** 0: Disable interrupt 1: Enable interrupt

1B101A20 QDMA_INT_GRP1 QDMA Interrupt Group 1 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_GRP1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_INT_GRP1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QDMA_INT_GRP1	**Interrupt group 1 assignment.** Each bit's definition is same as "QDMA_INT_STS" 0: Leave to QDMA int group 0, if (QDMA_INT_GRP1[n]==0 & QDMA_INT_GRP2[n]==0) 1: Assign to QDMA int group 1

1B101A24 QDMA_INT_GRP2 QDMA Interrupt Group 2 Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_GRP2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_INT_GRP2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description

31:0	QDMA_INT_GRP2	**Interrupt group 2 assignment.** Each bit's definition is same as "QDMA_INT_STS" 0: Leave to QDMA int group 0, if (QDMA_INT_GRP1[n]==0 & QDMA_INT_GRP2[n]==0) 1: Assign to QDMA int group 2

1B101A2C QDMA_DROP_PREC QDMA Drop by FFA Percentage Control 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SW_DROP_PREC															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HW_DROP_PREC															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	SW_DROP_PREC	**QDMA SW Drop by FFA Percentage Control** QDMA drop packets based on the full percertage of Free-for-All [31:16] = [PQ#15:PQ#0] 0: Disable drop by FFA percentage 1: Enable drop by FFA percentage
15:0	HW_DROP_PREC	**QDMA HW Drop by FFA Percentage Control** QDMA drop packets based on the full percertage of Free-for-All [15:0] = [PQ#15:PQ#0] 0: Disable drop by FFA percentage 1: Enable drop by FFA percentage

1B101A40 QDMA_HRED1 QDMA HW RED Distribution - I FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FFA_LOW_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FFA_MIDDLE_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:16	FFA_LOW_UTIL	**Low Utilization of FFA on HW path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 25% probability.
15:0	FFA_MIDDLE_UTIL	**Middle Utilization of FFA on HW path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 50% probability.

1B101A44 QDMA_HRED2 QDMA HW RED Distribution - II FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	REV0															FFA_UTIL_SEL
Type	DC															RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FFA_HIGH_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:17	REV0	**Reserved**
16	FFA_UTIL_SEL	**Manually Set FFA Utilization** When the bit is set, FFA Utilization will be configurable for user.
15:0	FFA_HIGH_UTIL	**High Utilization of FFA on HW path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 75% probability.

1B101A48 QDMA_SRED1 QDMA SW RED Distribution - I FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FFA_LOW_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FFA_MIDDLE_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:16	FFA_LOW_UTIL	**Low Utilization of FFA on HW path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 25% probability.
15:0	FFA_MIDDLE_UTIL	**Middle Utilization of FFA on CPU path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 50% probability.

1B101A4C QDMA_SRED2 QDMA SW RED Distribution - II FFFFFFFF

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0															FFA_UTIL_SEL
Type	DC															RW
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FFA_MIDDLE_UTIL															
Type	RW															
Reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit(s)	Name	Description
31:17	REV0	**Reserved**
16	FFA_UTIL_SEL	**Manually Set FFA Utilization**

When the bit is set, FFA Utilization will be configurable for user.

15:0	FFA_MIDDLE_UTIL	**High Utilization of FFA on CPU path** When the utilization of FFA reaches the threshold, RED will start to drop the en-queued packets by 75% probability.

1B101A50 QTX_Q0MIN_BKQDMA Queue #0 Min. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MIN_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MIN_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MIN_BUCKET	**The Token Value of the MIN Bucket.**

1B101A54 QTX_Q1MIN_BKQDMA Queue #1 Min. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MIN_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MIN_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MIN_BUCKET	**The Token Value of the MIN Bucket.**

1B101A58 QTX_Q2MIN_BKQDMA Queue #2 Min. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MIN_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MIN_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MIN_BUCKET	**The Token Value of the MIN Bucket.**

1B101A5C QTX_Q3MIN_BKQDMA Queue #3 Min. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	MIN_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MIN_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MIN_BUCKET	**The Token Value of the MIN Bucket.**

1B101A60 QTX_Q0MAX_BK QDMA Queue #0 Max. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAX_BUCKET	**The Token Value of the MAX Bucket.**

1B101A64 QTX_Q1MAX_BK QDMA Queue #1 Max. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAX_BUCKET	**The Token Value of the MAX Bucket.**

1B101A68 QTX_Q2MAX_BK QDMA Queue #2 Max. Bucket 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAX_BUCKET	**The Token Value of the MAX Bucket.**

1B101A6C **QTX_Q3MAX_BK** **QDMA Queue #3 Max. Bucket** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_BUCKET[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MAX_BUCKET[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	MAX_BUCKET	**The Token Value of the MAX Bucket.**

1B101A70 **QDMA_INT_STS_G0** **QDMA Interrupt Status Group 0** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_STS_G0[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_INT_STS_G0[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QDMA_INT_STS_G0	**Interrupt group 0 status, this information already "and" with "QDMA_INT_IMR"** Each bit definition is same as "QDMA_INT_STS"

1B101A74 **QDMA_INT_STS_G1** **QDMA Interrupt Status Group 1** **00000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_STS_G1[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_INT_STS_G1[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QDMA_INT_STS_G1	**Interrupt group 1 status, this information already "and" with**

"QDMA_INT_IMR"
Each bit definition is same as "QDMA_INT_STS"

1B101A78 QDMA_INT_STS_G2 QDMA Interrupt Status Group 2 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QDMA_INT_STS_G2[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_INT_STS_G2[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QDMA_INT_STS_G2	**Interrupt group 2 status, this information already "and" with "QDMA_INT_IMR"** Each bit definition is same as "QDMA_INT_STS"

1B101AD0 QTX_MIB_PCNT TX Queue MIB Forward Packet Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QTX_MIB_PCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QTX_MIB_PCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QTX_MIB_PCNT	**Accumulated forwarded packet count, according to VQTX_MIB_QID.** If MIB_PQ_SHW_SEP = 1, this counter will be seperated into two 16bits counter, [31:16] for software path, [15:0] for hardware path.

1B101AD4 QTX_MIB_DPCNT TX Queue MIB Dropped Packet Count 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	QTX_MIB_DPCNT[31:16]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QTX_MIB_DPCNT[15:0]															
Type	RO															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	QTX_MIB_DPCNT	**Accumulated dropped packet count, according to VQTX_MIB_QID.** If MIB_PQ_SHW_SEP = 1, this counter will be seperated into two 16bits

counter, [31:16] for software path, [15:0] for hardware path.

1B101B00 QTX_CTX_PTR TX Forward CPU Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FWD_CTX_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FWD_CTX_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FWD_CTX_PTR	**TX forward chain CPU pointer**

1B101B04 QTX_DTX_PTR TX Forward DMA Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FWD_DTX_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FWD_DTX_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FWD_DTX_PTR	**Tx forward chain DMA pointer** When QDMA TX is disabled, this field can be set to the initial pointer by CPU. After QDMA TX is enabled, this field is read-only.

1B101B08 QTX_FWD_CNT TX Forward DMA Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	ACC_HW_CNT															
Type	RC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACC_SW_CNT															
Type	RC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	ACC_HW_CNT	**Accumulated HW forwarded counter**
15:0	ACC_SW_CNT	**Accumulated SW forwarded counter**

1B101B10 QTX_CRX_PTR TX Release CPU Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16

Name	RLS_CRX_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RLS_CRX_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RLS_CRX_PTR	**TX release chain CPU pointer**

1B101B14 QTX_DRX_PTR TX Release DMA Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RLS_DRX_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RLS_DRX_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	RLS_DRX_PTR	**Tx release chain DMA pointer** When QDMA TX is disabled, this field can be set to the initial pointer by CPU. After QDMA TX is enabled, this field is read-only.

1B101B18 QTX_RLS_CNT TX Release DMA Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	ACC_RLS_CNT															
Type	RC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	REV0	**Reserved**
15:0	ACC_RLS_CNT	**Accumulated TX released descriptor count**

1B101B20 QDMA_FQ_HEAD Free Page Head Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FQ_HEAD_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FQ_HEAD_PTR[15:0]															
Type	RW															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FQ_HEAD_PTR	**Free buffer head pointer** When QDMA TX is disabled, this field can be set to the initial pointer by CPU. After QDMA TX is enabled, this field is read-only.

1B101B24 QDMA_FQ_TAIL Free Page Tail Pointer 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	FQ_TAIL_PTR[31:16]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FQ_TAIL_PTR[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:0	FQ_TAIL_PTR	**Free buffer tail pointer** When QDMA TX is diabled, this field can be set to the initial pointer by CPU. After QDMA TX is enabled, this field is read-only.

1B101B28 QDMA_FQ_CNT Free Page Counter 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	SWFQ_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	HWFQ_CNT															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:16	SWFQ_CNT	**Free buffer on CPU Pool** Initial free buffer count on CPU pool when QDMA_GLO_CFG.TX_DMA_EN is de-asserted.
15:0	HWFQ_CNT	**Free buffer on HW Pool** Initial free buffer count on HW pool when QDMA_GLO_CFG.TX_DMA_EN is de-asserted.

1B101B2C QDMA_FQ_BLEN Free Page Buffer Length 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	REV0		FQ_BUF_BLEN													
Type	DC		RW													
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	REV1															

Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:30	REV0	
29:16	FQ_BUF_BLEN	**Free buffer byte length** Configured free buffer length, QDMA RX will fetch the length info to decide the payload length on one descriptor/buffer.
15:0	REV1	

1B101B80 QDMA_RATE_EXP0 QDMA Scheduler Rate Exponent 0 000F4240

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP0[19:16]			
Type													RW			
Reset													1	1	1	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP0[15:0]															
Type	RW															
Reset	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP0	**Exponent part of the QDMA scheduler rate control** Rate(bps) = clock_frequency*8*(RATE_MAN/RATE_EXP) Bucket clock frequency is based on the setting of BKTCLK_ASYNC. BKTCLK_ASYNC = 0, bucket clock is system bus clock BKTCLK_ASYNC = 1, bucket clock is a dedicated 125Mhz clock 20'hF4240: Provide 1kbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

1B101B84 QDMA_RATE_EXP1 QDMA Scheduler Rate Exponent 1 000186A0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP1[19:16]			
Type													RW			
Reset													0	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP1[15:0]															
Type	RW															
Reset	1	0	0	0	0	1	1	0	1	0	1	0	0	0	0	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP1	**Exponent part of the QDMA scheduler rate control** 20'h186A0: Provide 10kbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

1B101B88 QDMA_RATE_EXP2 QDMA Scheduler Rate Exponent 2 00002710

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP2[19:16]			
Type													RW			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP2[15:0]															
Type	RW															
Reset	0	0	1	0	0	1	1	1	0	0	0	1	0	0	0	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP2	**Exponent part of the QDMA scheduler rate control** 20'h02710: Provide 100kbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

1B101B8C QDMA_RATE_EXP3 QDMA Scheduler Rate Exponent 3 000003E8

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP3[19:16]			
Type													RW			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP3[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP3	**Exponent part of the QDMA scheduler rate control** 20'h003E8: Provide 1Mbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

1B101B90 QDMA_RATE_EXP4 QDMA Scheduler Rate Exponent 4 00000064

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP4[19:16]			
Type													RW			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP4[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP4	**Exponent part of the QDMA scheduler rate control** 20'h00064: Provide 10Mbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

1B101B94 QDMA_RATE_EXP5 QDMA Scheduler Rate Exponent 5 0000000A

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name													QDMA_RATE_EXP5[19:16]			
Type													RW			
Reset													0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	QDMA_RATE_EXP5[15:0]															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

Bit(s)	Name	Description
19:0	QDMA_RATE_EXP5	**Exponent part of the QDMA scheduler rate control** 20'h0000A: Provide 100Mbps rate control (default setting with BKTCLK_ASYNC=1 and RATE_MAN=1)

23.4.1 Frame Engine MIB Counters

Base Address: 1B00_0000 (ex: PPE_AC_BCNT0 address is 1B00_2000)

Counter Table			
Offset	**Name**	**Width**	**Description**
0x2000	PPE_AC_BCNT0	64	PPE Accounting Group #0 Byte Counter
0x2008	PPE_AC_PCNT0	32	PPE Accounting Group #0 Packet Counter
0x200C	PPE_MTR_CNT0	32	PPE Meter Group #0
0x2010	PPE_AC_BCNT1	64	PPE Accounting Group #1 Byte Counter
0x2018	PPE_AC_PCNT1	32	PPE Accounting Group #1 Packet Counter
0x201C	PPE_MTR_CNT1	32	PPE Meter Group #1
.....	.....		
0x23F0	PPE_AC_BCNT63	64	PPE Accounting Group #63 Byte Counter
0x23F8	PPE_AC_PCNT63	32	PPE Accounting Group #63 Packet Counter
0x23FC	PPE_MTR_CNT63	32	PPE Meter Group #63
0x2400	GDM1_RX_GBCNT	64	Received good byte count for GDM1
0x2408	GDM1_RX_GPCNT	32	Received good packet count for GDM1
0x240C	-	32	Reserved
0x2410	GDM1_RX_OERCNT	32	Recevied overflow error packet count for GDM1
0x2414	GDM1_RX_FERCNT	32	Recevied FCS error packet count for GDM1
0x2418	GDM1_RX_SERCNT	32	Recevied too short error packet count for GDM1
0x241C	GDM1_RX_LERCNT	32	Recevied too long error packet count for GDM1
0x2420	GDM1_RX_CERCNT	32	Recevied IP/TCP/UDP checksum error packet count for GDM1
0x2424	GDM1_RX_FCCNT	32	Recevied flow control packet count for GDM1
0x2428	GDM1_TX_SKIPCNT	32	Transmit abort count for GDM1
0x242C	GDM1_TX_COLCNT	32	Transmit collision count for GDM1

0x2430	GDM1_TX_GBCNT	64	Trasnmit good byte count for GDM1
0x2438	GDM1_TX_GPCNT	32	Transmit good packet count for GDM1 (exlcude flow control frames)
0x243C	-	32	Reserved
0x2440	GDM2_RX_GBCNT	64	Received good byte count for GDM2
0x2448	GDM2_RX_GPCNT	32	Received good packet count for GDM2
0x244C	-	32	Reserved
0x2450	GDM2_RX_OERCNT	32	Recevied overflow error packet count for GDM2
0x2454	GDM2_RX_FERCNT	32	Recevied FCS error packet count for GDM2
0x2458	GDM2_RX_SERCNT	32	Recevied too short error packet count for GDM2
0x245C	GDM2_RX_LERCNT	32	Recevied too long error packet count for GDM2
0x2460	GDM2_RX_CERCNT	32	Recevied IP/TCP/UDP checksum error packet count for GDM2
0x2464	GDM2_RX_FCCNT	32	Recevied flow control packet count for GDM2
0x2468	GDM2_TX_SKIPCNT	32	Transmit abort count for GDM2
0x246C	GDM2_TX_COLCNT	32	Transmit collision count for GDM2
0x2470	GDM2_TX_GBCNT	64	Trasnmit good byte count for GDM2
0x2478	GDM2_TX_GPCNT	32	Transmit good packet count for GDM2 (exlcude flow control frames)
0x247C	-	32	Reserved

MAC_PPSC **PHY Polling and SMI Master Control** **05000000**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PHY_AP_EN	PHY_PRE_EN	PHY_MDC_CFG						RESV0			MDC_TURBO	RESV1		EE_AN_EN	
Type	RW	RW	RW						DC			RW	DC		RW	
Reset	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV2			PHY_END_ADDR					RESV3			PHY_START_ADDR				
Type	DC			RW					DC			RW				
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	PHY_AP_EN	**PHY auto polling enable** Indicate all PHY's status will be updated to PHY's status regsiters by PHY polling process.
30	PHY_PRE_EN	**PHY preample enable** Indicates SMI master will send preamble bits (32-bits) at each MDIO read/write transaction. [note] This bit will affect both PHY polling mode and PHY

Bit(s)	Name	Description
		indirect access mode.
29:24	PHY_MDC_CFG	**PHY MDC clock configuration** Used to configure the divider N for MDC clock frequency. MDC clock is sourced from 12.5MHz system clock and divided by N. [note] MDC clock is gated or disabled when PHY_MDC_CFG is set to 0.
23:21	RESV0	**Reserved**
20	MDC_TURBO	**MDC clock Turbo mode** When this bit is set, MDC clock is sourced from 25MHz system clock and divided by PHY_MDC_CFG.
19:18	RESV1	**Reserved**
17:16	EE_AN_EN	**PHY EEE auto-polling enable**
15:13	RESV2	**Reserved**
12:8	PHY_END_ADDR	**PHY Ppolling end address** Indicate the end of PHY address of PHY auto-polling process. [note] The difference between and start and end must be 1.
7:5	RESV3	**Reserved**
4:0	PHY_START_ADDR	**PHY Ppolling start address** Indicate the start of PHY address of PHY auto-polling process.

1B110004 MAC_PIAC PHY Indirect Access Control 00090000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	PHY_ACS_ST	RESV0	MDIO_REG_ADDR					MDIO_PHY_ADDR					MDIO_CMD		NMDIO_ST	
Type	W1C	DC	RW					RW					RW		RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	MDIO_RW_DATA															
Type	RW															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31	PHY_ACS_ST	**PHY access start** Start to indirect access PHY's register. While PHY's register access is complete, this bit will be self-cleared to 0. 0: Idle or indirect access complete 1: Start PHY access
30	RESV0	**Reserved**
29:25	MDIO_REG_ADDR	**Register Address (Clause 22) or Device Address (Clause 45)**
24:20	MDIO_PHY_ADDR	**PHY Address (Clause 22) or Port Address (Clause 45)**
19:18	MDIO_CMD	**MDIO command** 2'b00: Address (Clause 45) 2'b01: MDIO write 2'b10: MDIO read (Clause 22) / Read inc (Clause 45) 2'b11: MDIO read (Clause 45)
17:16	NMDIO_ST	**MDIO Start Field** 2'b00: Start (Clause 45) 2'b01: Start (Clause 22) other: Reserved
15:0	MDIO_RW_DATA	**MDIO Read/Write data** This is used as MDIO data field for read/write data access. When MDIO write command is activated, this is used as MDIO write data field. When MDIO read command is activated, this is used as MDIO read data field for read access only.

1B110008 MAC_GPC GIGA MAC Port Control 000C000C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0								P2_RMII_CKRX	P2_RMII_CKIN	P2_TMII_FREQ	P2_TMII_MODE	P2_TX_CLK_MODE	P2_RX_CLK_MODE	P2_RX_SKEW	
Type	DC								RW	RW	RW	RW	RW	RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESV1										P1_TMII_FREQ	P1_TMII_MODE	P1_TX_CLK_M	P1_RX_CLK_M	P1_RX_SKEW	

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
													ODE	ODE		
Type	DC										RW	RW	RW	RW	RW	
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0

Bit(s)	Name	Description
31:24	RESV0	**Reserved**
23	P2_RMII_CKRX	**RMII Reference Clock Pin Selection** 0: TXC 1: RXC
22	P2_RMII_CKIN	**RMII Reference Clock Direction** 0: Output 1: Input
21	P2_TMII_FREQ	
20	P2_TMII_MODE	**TMII mode** Switch to Turbot MII mode
19	P2_TX_CLK_MODE	**P2 TX clcok control** 0: HP mode (clock and data are in-phase) 1: 3Com mode (clock and data is 90 degree offset)
18	P2_RX_CLK_MODE	**P2 RX clock control** 0: delay 2ns on input rx_clk 1: no delay
17:16	P2_RX_SKEW	**P2 RX clock skew control** 2'b00: no dealy 2'b01: delay 150ps 2'b10: dealy 300ps 2'b11: clock inversion
15:6	RESV1	**Reserved**
5	P1_TMII_FREQ	
4	P1_TMII_MODE	**TMII mode** Switch to Turbot MII mode
3	P1_TX_CLK_MODE	**P1 TX clcok control** 0: HP mode (clock and data are in-phase) 1: 3Com mode (clock and data is 90 degree offset)
2	P1_RX_CLK_MODE	**P1 RX clock control** 0: delay 2ns on input rx_clk 1: no delay
1:0	P1_RX_SKEW	**P1 RX clock skew control** 2'b00: no dealy 2'b01: delay 150ps 2'b10: dealy 300ps 2'b11: clock inversion

1B110100 MAC_P1_MCR Port 1 MAC Control 20006300

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_RX_JUMBO				RESV0		MAC_RX_PKT_LEN		MTCC_LMT				IPG_CFG		RESV1	
Type	RW				DC		RW		RW				RW		DC	

e																
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FORCE_MODE	MAC_TX_EN	MAC_RX_EN	RESV2		PRMBL_LMT_EN	BKOFF_EN	BACKPR_EN	FORCE_EEE1G	FORCE_EEE10o	FORCE_RX_FC	FORCE_TX_FC	FORCE_SPD		FORCE_DPX	FORCE_LINK
Type	RW	RW	RW	DC		RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	MAX_RX_JUMBO	**Maximum received jumbo packet length** 4'h0, 4'h1: Reserved 4'h2: 2 Kbytes (maxi. length on FE/GDM) 4'h3 .. 4'hF: Reserved
27:26	RESV0	
25:24	MAC_RX_PKT_LEN	**Maximum received packet length** Sets the maximum length of ingress packets including CRC that can be received by MAC. 00: 1518 bytes for untagged frames; 1522 bytes for tagged frames 01: 1536 bytes 10: 1552 bytes 11: MAX_RX_JUMBO
23:20	MTCC_LMT	**MTCC limit** Maximum transmit collision count limitation. 0: Disable TX collision abort function, send packet until successfully 1 ~ 15: Maximum transmit collision count
19:18	IPG_CFG	**Inter-Frame Gap Shrink** 2'b00: normal 96-bits IFG 2'b01: Transmit 96-bits IFG with short IFG in random behavior. 2'b10: shrink 64-bits IFG 2'b11: When any output queue inside the port is congested, shrink 64-bits IFG will be enabled; otherwise, normal 96-bits IFG will be the default.
17:16	RESV1	**Reserved**
15	FORCE_MODE	**Force MAC Mode**

Bit(s)	Name	Description
		0: force mode off (MAC status is decided by PHY auto-polling) 1: force mode on (MAC status is determined by FORCE_XXX register)
14	MAC_TX_EN	**TX MAC Enable** 0: TX MAC function is disabled 1: TX MAC function is enabled
13	MAC_RX_EN	**RX MAC Enable** 0: RX MAC function is disabled 1: RX MAC function is enabled
12:11	RESV2	**Reserved**
10	PRMBL_LMT_EN	**Preamble Limit Enable** 0: RXMAC can recognize the Start Frame Delimiter (SFD), without needing to receive a byte with the value of 55 in the preamble. 1: RXMAC will recognize the SFD before the next new frame when it receives the 7 consecutive bytes with the value of 55 within the 8-byte Preamble. If SFD (8'hd5) shows up after the 8-byte Preamble, RXMAC will not recognize it and treat it as if there were no SFD.
9	BKOFF_EN	**Back-off Enable** 0: disabled 1: Let MAC follow the back-off mechanism when collision happens
8	BACKPR_EN	**Back-Pressure enable** 0: disabled 1: Enable back-pressure mechanism when operating in half-duplex mode and flow-control is on
7	FORCE_EEE1G	**Force LPI mode for 1000m link speed** When (FORCE_MODE= 1), these bits are used to control MAC EEE mode 0: Donot have the ability of entering EEE low power idle mode for 1000m link speed 1: Have the ability of entering EEE low power idle mode
6	FORCE_EEE100	**Force LPI mode for 100m link speed** When (FORCE_MODE= 1), these bits are used to control MAC EEE mode 0: Donot have the ability of entering EEE low power idle mode for 100m link speed 1: Have the ability of entering EEE low power idle mode
5	FORCE_RX_FC	**Force RX flow cotnrol** When (FORCE_MODE= 1), these bits are used to control MAC

Bit(s)	Name	Description
		RX flow control 0: Disable 1: Let MAC accept the pause frame when operating in full-duplex mode and flow-control is on
4	FORCE_TX_FC	**Force Tx flow cotnrol** When (FORCE_MODE= 1), these bits are used to control MAC TX flow control 0: Disable 1: Let MAC transmit the pause frame when operating in full-duplex mode and flow-control is on
3:2	FORCE_SPD	**Force link speed** When (FORCE_MODE= 1), these bits are used to control MAC link speed 2'b00: 10mbps 2'b01: 100mbps 2'b10: 1000mbps 2'b11: reserved
1	FORCE_DPX	**Force duplex** When (FORCE_MODE= 1), these bits are used to control MAC link duplex 0: Half-duplex 1: Full-duplex
0	FORCE_LINK	**Force link state** When (FORCE_MODE= 1), these bits are used to control MAC link state 0: Link down 1: Link up

1B110104 MAC_P1_EEE Port 1 EEE Control 111E1E02

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WAKEUP_TIME_1000								WAKEUP_TIME_100							
Type	RW								RW							
Reset	0	0	0	1	0	0	0	1	0	0	0	1	1	1	1	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LPI_TXIDLE_THD								RESVo				CKG_TXIDLE	CKG_RXLPI	TX_DOWN_REQ	LPI_MODE
Type	RW								RW				RW	RW	RW	RW
Reset	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1	0

Bit(s)	Name	Description

Bit(s)	Name	Description
31:24	WAKEUP_TIME_1000	**Wake up time for 1000mbps LIP mode** The minimum allowed time needed to wait for PHY to be fully functional and TXMAC can transmit packet after wake-up Time unit: 1us
23:16	WAKEUP_TIME_100	**Wake up time for 100mbps LIP mode** The minimum allowed time needed to wait for PHY to be fully functional and TXMAC can transmit packet after wake-up Time unit: 1us
15:8	LPI_TXIDLE_THD	**TX IDLE timer to enter LPI mode** When there is no packet to be transmitted and exceeds time period specified by LPI_TXIDLE_THD, the TXMAC will automatically enter LPI (Low Power Idle) mode and send EEE LPI signal to link partner Time unit: 1ms
7:4	RESV0	**Reserved**
3	CKG_TXIDLE	**TX Clock Power Down** 0: disable 1: Stop TX clock when the corresponding port has no traffic to send and has entered IDLE state for <LPI_TXIDLE_THD> ms.
2	CKG_RXLPI	**RX Clock Power Down** 0: disable 1: Stop RX clock ticking when the corresponding port entering the LPI mode and IDLE state.
1	TX_DOWN_REQ	**TX DMA Request** Keep TX request when this port is linked down 0: No TX request 1: Keep TX request when this port is linked down.
0	LPI_MODE	**LPI mode activated** 0: DIsable 1: Let the system enter LPI mode immediatedly and send EEE LPI frame to the link partner

1B110108 MAC_P1_SR Port 1 MAC Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESV0															
Type	DC															

Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESVo								EEE1G_STATUS	EEE100_STATUS	RX_FC_STATUS	TX_FC_STATUS	SPD_STATUS		DPX_STATUS	LINK_STATUS
Type	DC								RO	RO	RO	RO	RO		RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RESVo	**Reserved**
7	EEE1G_STATUS	**LPI mode for 1000m link speed** 0: Donot have the ability of entering EEE low power idle mode for 1000m link speed 1: Have the ability of entering EEE low power idle mode
6	EEE100_STATUS	**LPI mode for 100m link speed** 0: Donot have the ability of entering EEE low power idle mode for 100m link speed 1: Have the ability of entering EEE low power idle mode
5	RX_FC_STATUS	**RX flow cotnrol status** 0: Disable 1: Let MAC accept the pause frame when operating in full-duplex mode and flow-control is on
4	TX_FC_STATUS	**TX flow cotnrol status** 0: Disable 1: Let MAC transmit the pause frame when operating in full-duplex mode and flow-control is on
3:2	SPD_STATUS	**Link speed status** 2'b00: 10mbps 2'b01: 100mbps 2'b10: 1000mbps 2'b11: reserved
1	DPX_STATUS	**Duplex status** 0: Half-duplex 1: Full-duplex

Bit(s)	Name	Description
0	LINK_STATUS	**Link state** 0: Link down 1: Link up

1B110200 MAC_P2_MCR Port 2 MAC Control 20006300

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	MAX_RX_JUMBO				RESV0		MAC_RX_PKT_LEN		MTCC_LMT				IPG_CFG		RESV1	
Type	RW				DC		RW		RW				RW		DC	
Reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	FORCE_MODE	MAC_TX_EN	MAC_RX_EN	RESV2		PRMBL_LMT_EN	BKOFF_EN	BACKPR_EN	FORCE_EEE1G	FORCE_EEE100	FORCE_RX_FC	FORCE_TX_FC	FORCE_SPD		FORCE_DPX	FORCE_LINK
Type	RW	RW	RW	DC		RW	RW	RW	RW	RW	RW	RW	RW		RW	RW
Reset	0	1	1	0	0	0	1	1	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:28	MAX_RX_JUMBO	**Maximum received jumbo packet length** 4'h0, 4'h1: Reserved 4'h2: 2 Kbytes (maxi. length on FE/GDM) 4'h3 .. 4'hF: Reserved
27:26	RESV0	
25:24	MAC_RX_PKT_LEN	**Maximum received packet length** Sets the maximum length of ingress packets including CRC that can be received by MAC. 00: 1518 bytes for untagged frames;1522 bytes for tagged frames 01: 1536 bytes 10: 1552 bytes 11: MAX_RX_JUMBO
23:20	MTCC_LMT	**MTCC limit** Maximum transmit collision count limitation. 0: Disable TX collision abort function, send packet until

Bit(s)	Name	Description
		successfully 1 ~ 15: Maximum transmit collision count
19:18	IPG_CFG	**Inter-Frame Gap Shrink** 2'b00: normal 96-bits IFG 2'b01: Transmit 96-bits IFG with short IFG in random behavior. 2'b10: shrink 64-bits IFG 2'b11: When any output queue inside the port is congested, shrink 64-bits IFG will be enabled; otherwise, normal 96-bits IFG will be the default.
17:16	RESV1	**Reserved**
15	FORCE_MODE	**Force MAC Mode** 0: force mode off (MAC status is decided by PHY auto-polling) 1: force mode on (MAC status is determined by FORCE_XXX register)
14	MAC_TX_EN	**TX MAC Enable** 0: TX MAC function is disabled 1: TX MAC function is enabled
13	MAC_RX_EN	**RX MAC Enable** 0: RX MAC function is disabled 1: RX MAC function is enabled
12:11	RESV2	**Reserved**
10	PRMBL_LMT_EN	**Preamble Limit Enable** 0: RXMAC can recognize the Start Frame Delimiter (SFD), without needing to receive a byte with the value of 55 in the preamble. 1: RXMAC will recognize the SFD before the next new frame when it receives the 7 consecutive bytes with the value of 55 within the 8-byte Preamble. If SFD (8'hd5) shows up after the 8-byte Preamble, RXMAC will not recognize it and treat it as if there were no SFD.
9	BKOFF_EN	**Back-off Enable** 0: disabled 1: Let MAC follow the back-off mechanism when collision happens
8	BACKPR_EN	**Back-Pressure enable** 0: disabled 1: Enable back-pressure mechanism when operating in half-duplex mode and flow-control is on
7	FORCE_EEE1G	**Force LPI mode for 1000m link speed**

Bit(s)	Name	Description
		When (FORCE_MODE= 1), these bits are used to control MAC EEE mode 0: Donot have the ability of entering EEE low power idle mode for 1000m link speed 1: Have the ability of entering EEE low power idle mode
6	FORCE_EEE100	**Force LPI mode for 100m link speed** When (FORCE_MODE= 1), these bits are used to control MAC EEE mode 0: Donot have the ability of entering EEE low power idle mode for 100m link speed 1: Have the ability of entering EEE low power idle mode
5	FORCE_RX_FC	**Force RX flow cotnrol** When (FORCE_MODE= 1), these bits are used to control MAC RX flow control 0: Disable 1: Let MAC accept the pause frame when operating in full-duplex mode and flow-control is on
4	FORCE_TX_FC	**Force Tx flow cotnrol** When (FORCE_MODE= 1), these bits are used to control MAC TX flow control 0: Disable 1: Let MAC transmit the pause frame when operating in full-duplex mode and flow-control is on
3:2	FORCE_SPD	**Force link speed** When (FORCE_MODE= 1), these bits are used to control MAC link speed 2'b00: 10mbps 2'b01: 100mbps 2'b10: 1000mbps 2'b11: reserved
1	FORCE_DPX	**Force duplex** When (FORCE_MODE= 1), these bits are used to control MAC link duplex 0: Half-duplex 1: Full-duplex
0	FORCE_LINK	**Force link state** When (FORCE_MODE= 1), these bits are used to control MAC link state 0: Link down 1: Link up

1B110204 MAC_P2_EEE Port 2 EEE Control 00001E02

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WAKEUP_TIME_1000								WAKEUP_TIME_100							
Type	RW								RW							
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	LPI_TXIDLE_THD								RESV0				CKG_TXIDLE	CKG_RXLPI	TX_DOWN_REQ	LPI_MODE
Type	RW								DC				RW	RW	RW	RW
Reset	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1	0

Bit(s)	Name	Description
31:24	WAKEUP_TIME_1000	**Wake up time for 1000mbps LIP mode** The minimum allowed time needed to wait for PHY to be fully functional and TXMAC can transmit packet after wake-up Time unit: 1us
23:16	WAKEUP_TIME_100	**Wake up time for 100mbps LIP mode** The minimum allowed time needed to wait for PHY to be fully functional and TXMAC can transmit packet after wake-up Time unit: 1us
15:8	LPI_TXIDLE_THD	**TX IDLE timer to enter LPI mode** When there is no packet to be transmitted and exceeds time period specified by LPI_TXIDLE_THD, the TXMAC will automatically enter LPI (Low Power Idle) mode and send EEE LPI signal to link partner Time unit: 1ms
7:4	RESV0	**Reserved**
3	CKG_TXIDLE	**TX Clock Power Down** 0: disable 1: Stop TX clock when the corresponding port has no traffic to send and has entered IDLE state for <LPI_TXIDLE_THD> ms.
2	CKG_RXLPI	**RX Clock Power Down** 0: disable 1: Stop RX clock ticking when the corresponding port entering the LPI mode and IDLE state.
1	TX_DOWN_REQ	**TX DMA Request** Keep TX request when this port is linked down 0: No TX request 1: Keep TX request when this port is linked down.

Bit(s)	Name	Description
0	LPI_MODE	**LPI mode activated** 0: Disable 1: Let the system enter LPI mode immediatedly and send EEE LPI frame to the link partner

1B110208 MAC_P2_SR Port 2 MAC Status 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	RESVo															
Type	DC															
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESVo								EEE1G_STATUS	EEE100_STATUS	RX_FC_STATUS	TX_FC_STATUS	SPD_STATUS		DPX_STATUS	LINK_STATUS
Type	DC								RO	RO	RO	RO	RO		RO	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:8	RESVo	**Reserved**
7	EEE1G_STATUS	**LPI mode for 1000m link speed** 0: Donot have the ability of entering EEE low power idle mode for 1000m link speed 1: Have the ability of entering EEE low power idle mode
6	EEE100_STATUS	**LPI mode for 100m link speed** 0: Donot have the ability of entering EEE low power idle mode for 100m link speed 1: Have the ability of entering EEE low power idle mode
5	RX_FC_STATUS	**RX flow cotnrol status** 0: Disable 1: Let MAC accept the pause frame when operating in full-duplex mode and flow-control is on
4	TX_FC_STATUS	**TX flow cotnrol status** 0: Disable 1: Let MAC transmit the pause frame when operating in full-

Bit(s)	Name	Description
		duplex mode and flow-control is on
3:2	SPD_STATUS	**Link speed status** 2'b00: 10mbps 2'b01: 100mbps 2'b10: 1000mbps 2'b11: reserved
1	DPX_STATUS	**Duplex status** 0: Half-duplex 1: Full-duplex
0	LINK_STATUS	**Link state** 0: Link down 1: Link up

1B110210 MAC_P2_WOL Port 2 WOL 00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	WOL_DBG														WOL_INT_CLR	WOL_STS
Type	RO														W1C	RO
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	RESVo												SNP_PKT	CRC_DIS	WOL_INT_EN	WOL_EN
Type	RO												RW	RW	RW	RW
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit(s)	Name	Description
31:18	WOL_DBG	**GMAC2 Wake-up On Lan Debug Signals**
17	WOL_INT_CLR	**GMAC2 Wake-up On Lan Interrupt Status**
16	WOL_STS	**GMAC2 Wake-up On Lan Status** If enable WOL_EN, this bit will change from 0 to 1 when GMAC RX state machine enter IDLE state. It indicates GMAC will drop all packets and detect magic packet. 0: Normal State 1: Listen State and detect magic packet

Bit(s)	Name	Description
15:4	RESV0	
3	SNP_PKT	**GMAC2 Wake-up On Lan with snoopy packet**
2	CRC_DIS	**GMAC2 Wake-up On Lan with CRC Check Disable**
1	WOL_INT_EN	**GMAC2 Wake-up On Lan Interrupt Enable**
0	WOL_EN	**GMAC2 Wake-up On Lan Function Enable**