

EXAMPLE OF COMMUNICATION WITH THE DEVICE AND ATTACHED CAN SLAVE

Saturday, 18 April, 2020 3:31 PM

```
gi //Get information command
0 cmd gi 0 NA StanislavSubrt CANMaster RevA 2.0.1 1 18.4.2020
d33c4e41 ack gi 0 0 31216
gh //Get help command
> CANMaster help.
> Space or tab interchangeably used as separator. Below in [] brackets is variable.
> Generic command syntax is: [CRC] [Cmd] [CmdId] [Param1] [Param2] ... [ParamN].
> Each command is acknowledged immediately: [CRC] ack [Cmd] [CmdId] [ErrorCode] [TimeStamp_ms]
> Events are asynchronous to command reception and acknowledge. CRC parameter is optional.
> Transmit message: [CRC] tx [CmdId] [MsgId] [ByteCnt] [B0] [B1] [B2] [B3] [B4] [B5] [B6] [B7].
> Answer is event: [CRC] evt canrx [TimeStamp_ms] [MsgId] [ByteCnt] [B0] [B1] [B2] [B3] [B4] [B5] [B6] [B7].
> Byte order might be changed for both tx command and canrx event: [CRC] so [CmdId] [Order 0=normal/1=inv].
> Currently set byte order might be read: [CRC] go [CmdId]. Response: [CRC] go [CmdId] [Order].
> Single next CAN transmission might be interrupted at specified delay time for specified duration
> using command: [CRC] si [CmdId] [Delay_us_uint16] [Duration_us_uint16].
> Please note that there is inherent 2us delay which adds to specified Delay_us_uint16.
> Enjoy the tool.
9ee1e5c8 ack gh 0 12 34907
tx 123456 8 04 00 00 00 00 00 00 //Transmit 8 bytes via CAN, first byte is 0x04
3d825459 ack tx 123456 0 195215 //Ack by canmaster
094f2094 evt err 195216 00001020 0000 0000 02 00 00 00 //Error event - transmission not successful
tx 123456 8 04 00 00 00 00 00 00 //Transmit again the samee
e01ce8b2 ack tx 123456 0 212990 //Acknowledged my canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit 8 bytes, first byte is 0x51
63070b78 ack tx 123456 0 304236 //Ack by canmaster
4c0b243c evt canrx 304237 080 8 51 60 00 00 00 00 00 //Response of attached CAN device 0x51 0x60 0x00...
go 123 //Get byte order
0 cmd go 123 0 //Response to above command - notice same CmdId. 0 indicates current byte order.
e20f66ca ack go 123 0 329778 //Ack by canmaster
so 123 1 //Set byte order
a08ed74b ack so 123 0 345114 //Ack by canmaster
go 1234 //Get byte order to verify success of previous command
0 cmd go 1234 1 //Response to above command, 1 indicates inverted byte order.
```

```

dac39a91 ack go 1234 0 353313 //Ack by canmaster
tx 123456 600 8 01 00 00 00 00 00 51 //Transmit message with inverted byte order where first byte is 0x01
797da331 ack tx 123456 0 438881 //Ack by canmaster
7a03c4c0 evt canrx 438882 080 8 00 00 00 00 00 60 51 //Response of attached CAN device
so 123 0 //Set byte order back to normal
4d94be79 ack so 123 0 453560 //Ack by canmaster
go 1234 //Verify byte order
0 cmd go 1234 0 //Byte order is normal
3f3041ee ack go 1234 0 457934 //Ack by canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit CAN message where first byte is 0x51
5d07416f ack tx 123456 0 469035 //Ack by canmaster
7a76d3ee evt canrx 469036 080 8 51 60 00 00 00 00 00 //Response of attached CAN device
si 123321 5 1 //Set CAN transmitted message interruption, 5us after transmission starts and duration of interruption is 1us
75621d26 ack si 123321 0 533797 //Ack by canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit the message with short interruption
c77ca697 ack tx 123456 0 545424 //Ack by canmaster
e43dfdc0 evt canrx 545425 080 8 51 60 00 00 00 00 00 //Response of attached CAN device is OK
si 123321 5 3 //Set CAN transmitted message interruption, 5us after transmission starts and duration of interruption is 3us
0f857a1e ack si 123321 0 567686 //Ack by canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit the message with short interruption
2751995b ack tx 123456 0 569943 //Ack by canmaster
996d74c0 evt canrx 569944 080 8 51 60 00 00 00 00 00 //Response of attached CAN device is OK
si 123321 5 5 //Set CAN transmitted message interruption, 5us after transmission starts and duration of interruption is 5us
8795e2b9 ack si 123321 0 583412 //Ack by canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit the message with short interruption
b868b11d ack tx 123456 0 587006 //Ack by canmaster
a324c449 evt canrx 587007 080 8 51 60 00 00 00 00 00 //Response of attached CAN device is OK
si 123321 5 6 //Set CAN transmitted message interruption, 5us after transmission starts and duration of interruption is 6us
55fb3436 ack si 123321 0 594604 //Ack by canmaster
tx 123456 600 8 51 00 00 00 00 00 01 //Transmit the message with short interruption
1d8dfeae ack tx 123456 0 597897 //Ack by canmaster
9c3caffa evt err 597897 00001080 0000 0000 00 00 00 00 //Error event - transmission not successful

```