Project Homathko

OBC and Software Design

Prepared for: UVic Homathko Design

Project Homathko

OBC and Software Design

Revision History

Rev	Effective Date	Author	Approver	Description of Changes
A	2016-05-03	Brosnan Yuen		Draft

1.0 Introduction

UVic's Satellite Design (UVSD) team is designing a cube-satellite called Project Homathko. Project Homathko will enter the Canadian Satellite Design Challenge (CSDC). Homathko contains On Board Computer (OBC). OBC is used for data processing on Homathko. This document describes the design of the OBC and software. The design will be broken up into a hardware section and a software section. The requirements will follow the Project Homathko system level and CSDC requirements.

2.0 Requirements

OBC and software subsystems have many requirements. They are split up into OBC hardware requirements and OBC software requirements.

2.1 OBC Hardware Requirements

Requirements for OBC hardware are listed below.

2.1.1 Functional Requirements

The table below lists the functions the OBC is required to perform.

Function	Description
Telemetry	OBC will obtain sensor data through STM32s. Sensor data will be logged or sent back to the ground station.
Payload control	OBC control the execution of the payload at a specific time. The OBC will record the attitude at that specific time.
Power control	OBC will monitor power consumption to prevent brown outs.
Data handling	OBC will process all data coming from all interfaces.
Timekeeping	OBC will sync the time on all of the MCUs.
Memory integrity	OBC will protect all read/write memory from errors.

Table 1. Functions of OBC.

2.1.2 Thermal-Vacuum Requirements

DIETR specifies a thermal-vacuum requirement [1]. OBC will withstand a vacuum of 5×10^{-4} Torr. OBC will also have an outgassing requirement of less than 1% of mass [2]. In addition, OBC will withstand a thermal cycle from -20 °C to 70 °C.

2.1.3 Mounting and PCB Requirements

PCB will be a 10 cm by 10 cm with an edge connector. Edge connector will have 100 pins connected to a Samtec Inc. MEC8-150-02-L-DV-A connector on the backplane. Power and system buses will be provided through the edge connectors.

2.1.4 Vibration Requirements

DIETR specifies a Launch Environment Tests requirement [3]. OBC will withstand a quasi-static acceleration test of 12g. The OBC will also withstand random vibrations of 0.1 g^2/Hz at 200 Hz.

2.1.5 Power Requirements

Power system provides several rails. However only the +3.5V input rail is used. Maximum current will be 2 A for the OBC. The voltages required by OBC are listed in Table 2.

Voltage(V)	Maximum voltage ripple(mV)	Maximum current(mA)
3.3	100	2000
2.5	1	10
1.3	70	20
1.2	80	440

Table 2. Required voltage rails.

2.1.6 Radiation Requirements

Homathko will be placed 800 km above earth in low earth orbit (LEO). Solar panels and 2 mm of aluminum will shield Homathko.

Type of radiation	Dose rate (MeV cm^2 g^-1)
Heavy ion	475.1912
Proton	962.1174
Electron	1358.5
Total	2824

Table 3. Dose rates [4].

The OBC will need to withstand the radiation dose rates listed above. The mission length is 2 years. Total ionization dose (TID) for entire duration is 2824 rad.

2.1.7 RF Shielding Requirements

There are radio boards on Homathko. The frequencies of radio boards are listed below.

Name	Frequency (MHz)

Communications TX	145.9
Communications RX	436
GPS L1	1575.42
GPS L2	1227.6
GPS B1	1561.098

Table 5. Radio frequencies.

The OBC needs to be shielded from the frequencies in Table 4 to reduce noise.

2.1.8 Mass Requirements

The mass of the OBC and the GPS will be less than 500g.

2.1.9 Memory Requirements

The nonvolatile long term memory storage needs to be at least 4MB. The memory needs to be protected from radiation. A copy of all the programs on Homathko must be stored in this memory.

2.1.10 Interface Requirements

OBC will need the following interfaces below.

Protocol	Rate (kbit/s)	QTY
I2C	400	1
SPI	16000	3
CAN bus ISO 11898-2	500	2
JTAG	100000	2
GPS PPS	10000	1
UART	115.2	2
EMIF	3656800	2

Table 5. Protocols and rates.

The interfaces shown in Table 5 are used to communicate between different ICs and systems. The most important protocol is the CAN bus. All data flows through the CAN bus.

2.1.11 Communication Requirements

The ADC and DAC needs to meet the communications requirements.

Component	Sample rate(kHz)	Resolution (bits)	Signal	Maximum Voltage (V)	Minimum Voltage (V)

ADC	198	16	Differential	3.3	0
DAC	198	12	Differential	3.3	0

Table 6. ADC and DAC specifications.

Table 6 shows the requirements for ADC and DAC. ADC and DAC are part of layer 1. The RF front end and DSP will implement layer 1 of the OSI model. Layer 1 is the physical layer for raw bit transmission.

2.1.12 Sensor Requirements

The OBC need the following sensors for operation:

- Magnetometer
- Gyroscope
- Accelerometer
- Temperature sensor
- GPS

The sensors are used for the attitude determination and control system (ADCS). Moreover, GPS is used for the location of the Homathko.

2.1.13 Hardware Watchdog Requirements

The OBC requires a dedicated hardware watchdog timer for the MCU. The DSP also requires a dedicated hardware watchdog timer. Watchdog timers must be active at all times.

2.2 OBC Software Requirements

Software requirements for the OBC are split up into many different sections.

2.2.1 Operating System Requirements

The OBC requires a lightweight and modular OS. The OS must function within a small microcontroller cache. The OS must have a real time control capabilities. The OS must be fault tolerant and be able to recover from data corruption.

2.2.2 File System Requirements

The file system must protect the data from radiation. File system must have multiple copies of the file system structure. File system must use error correction codes for data recovery. Data must be scrubbed automatically after a certain amount of time. File system must detect gate ruptures. In a event of a gate rupture the file system must relocate the data contained inside it. It also must avoid the irreparable bits due to the gate rupture.

2.2.3 System Data Bus Requirements

Data must be transmitted using system buses. The system buses must use a cyclic queue. Cyclic memory will avoid buffer overflows and out of range pointer indexes. The data must

have a timestamp accurate to 1 us and the sequence number. It also must have a cyclic redundancy check to ensure data integrity.

2.2.4 Watchdog Timer Requirement

In addition to hardware watchdog timers. The software must use software watchdog timers for monitoring tasks. For a time interval, the task must start a watchdog timer. The watchdog timer counts down until the watchdog timer is reset or reaches zero time. If the watchdog timer resets then the watchdog counts down from the top again. If the watchdog timer reaches zero time then the kernel must terminate the task and start the task again.

If the task runs outside of the TMS570 then the task must send a message to the TMS570 to start or reset the global watchdog timer. The task must receive an acknowledge after starting or resetting the timer. The task must also reset and start a local watchdog timer.

2.2.5 Communications Software Requirement

DSP will implement layer 2. Layer 2 is the data link layer for local transmission of frames. DSP will determine MAC, ARP, and LLC for the communications system. Layer 3 and above will be implemented in the microcontroller. Microcontroller will implement IPv4 for layer 3. Microcontroller will also have TCP for layer 5. TLS/SSL will be used for encrypted communications between ground station and Homathko.

2.3 Testing Requirements

Hardware needs to be test to meet the requirements above. Software requires automated testing and continuous integration. Near the end, a full systems test will be performed. After the full systems test, all components will be audited manually.

3.0 High Level Design

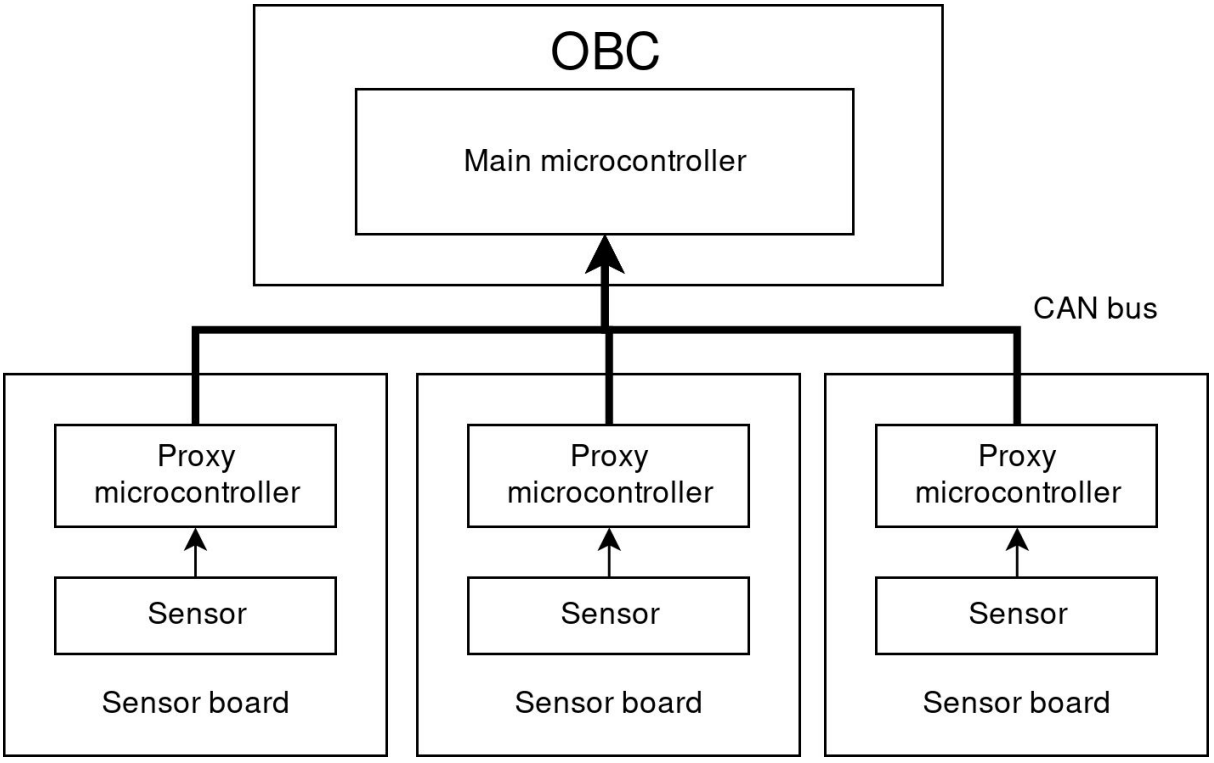

Figure 1. Overview of the system.

As shown in Figure 1, TMS570 is main microcontroller on the OBC. TMS570 is also the main microcontroller of Homathko. STM32s are the proxy microcontrollers on separate sensor boards. STM32s and TMS570 control data flow and processing. There are many sensors on the Homathko. A network of STM32s will be used to sample the sensors. Sensors feed data into the STM32s. Then STM32 passes the data to the TMS570 using CAN bus. All data goes to the TMS570 for processing and storage.

Some STM32s be tasked with controlling payload. TMS570 will send a command to the STM32 to execute the payload at specific time. STM32s makes sure the payload is functioning properly when the payload is executed. Some STM32s will be monitoring the power consumption of the Homathko. If the power consumption is too high then STM32s will send a warning message to TMS570. Then TMS570 will send a command back to STM32s for power regulation. STM32s may power off certain components to conserve power.

3.1 CAN Bus

STM32 and TMS570 microcontrollers communicate using the CAN bus. CAN bus is a bidirectional communication system. CAN bus uses ISO 11898-2 specification. ISO 11898-2 requires two 120 ohm termination resistors at the each end. The data rate is around 500kbps.

The CAN bus also serves a protocol to flash STM32s. TMS570 could write data directly to the STM32's flash using CAN bus. This allows TMS570 flash new programs onto the STM32. Moreover, TMS570 could scrub the memory regions inside the STM32s.

3.2 Silicone Potting

All PCBs will use silicon potting to increase reliability. Outer space environment increases probability of tin whiskers. Tin whiskers grows from tin based solder. Tin whiskers will cause short circuits between exposed solder joints. Short circuits could reduce the functionally of Homathko. Silicon potting prevents tin whiskers by coating the exposed tin solder. Silicon potting will also decrease mass outgassing in space. Moreover, silicon potting acts as a adhesive to ICs. Therefore, IC will tolerate shock and vibrations better.

4.0 Failure Mode and Effects Analysis

Failure Mode and Effects Analysis (FMEA) is used to measure the significance of possible failures. There are two types of FMEA, one before design and one after design.

FMEA Ref.	Item	Potential failure mode	Potential cause(s) / mechanism	Local effects of failure	System level effect
1	Memory	Data corruption	Particle radiation	Errors in computation and data loss	Multiple components may not function
2	Memory	Permanent hardware damage	Particle radiation	Local memory not usable	Reduced memory capacity
3	Microcontroller	Single Event Upset	Particle radiation	Errors in computation	System may function unexpectedly
4	Microcontroller	Single Event Effects	Particle radiation	Some functionality is lost	System functionality is lost
5	Communications	Interference from EM waves	EM radiation	Signal integrity degrades	Communications may receive incorrect data
6	Board connectors	Physical connection lost	Vibrations from rocket launch	Components will not function	Communications and power is lost
7	PCB	Short circuit	Tin whiskers	Components will not function	System functionality is lost
8	Communications	Insufficient SNR for DAC and ADC	Noise produced by power supply	Communications system efficiency is lost	Communications may receive incorrect data

9	Microcontroller	Overheating	Over consuming power	Components will not function	All functionality is lost
10	Sensor	Invalid data	Sensor failure, sensor installed incorrectly	Invalid satellite attitude and experiment data	Communications is lost and experiment fails
11	Microcontroller	Power supply failure	Component failure	All functionality is lost	All functionality is lost
12	Microcontroller	Underheating	Thermal design	All functionality is lost	All functionality is lost
13	Software	Time desynchronization	Clock drift	Sensor data has errors	Invalid satellite attitude and experiment data
14	Software	Race condition	Software bug	Microcontroller halts	System may function unexpectedly
15	Software	Fails to boot	Particle radiation	Microcontroller stops working	System functionality is lost
16	Software	System bus data corruption	Particle radiation	Data is invalid	System may function unexpectedly
17	Software	Filesystem corruption	Particle radiation	Data is invalid	System functionality is lost
18	Software	Memory leak	Software bug	Memory fails to be cleared	System may function unexpectedly
19	Software	Memory boundary error	Software bug	Data is corrupted	System may function unexpectedly
20	Software	Failure to handle errors	Software bug	Error continues	System may function unexpectedly

Table 7. Pre-design FMEA.

FMEA Ref.	Item	Potential failure mode	Probability	Severity	Detection	Risk Level	Detection Procedure
1	Memory	Data corruption	0.9999	0.9	0.2	1.07991	Radiate IC in test facility

2	Memory	Permanent hardware damage	0.2	0.95	0.2	0.38	Radiate IC in test facility
3	Microcontroller	Single Event Upset	0.2	0.9	0.2	0.36	Radiate IC in test facility
4	Microcontroller	Single Event Effects	0.01	0.99	0.2	0.2079	Radiate IC in test facility
5	Communications	Interference from EM waves	0.6	0.5	0.95	0.775	Simulate EM radiation using USRP
6	Board connectors	Physical connection lost	0.1	0.99	0.95	1.0395	Shake table testing
7	PCB	Short circuit	0.9	0.99	0.01	0.9009	Thermal vacuum test chamber
8	Communications	Insufficient SNR for DAC and ADC	0.2	0.2	0.99	0.238	Oscilloscope probing
9	Microcontroller	Overheating	0.1	0.99	0.2	0.297	Thermal vacuum test chamber
10	Sensor	Invalid sensor data	0.9	0.7	0.5	0.98	Sensor testing
11	Microcontroller	Power supply failure	0.1	0.99	0.2	0.297	Stress testing of power supply
12	Microcontroller	Underheating	0.1	0.99	0.2	0.297	Thermal vacuum test chamber
13	Software	Time desynchronization	0.3	0.2	0.4	0.14	Time measurement
14	Software	Race condition	0.9	0.5	0.3	0.6	Automated software testing

15	Software	Fails to boot	0.3	0.99	0.1	0.396	Radiate IC in test facility
16	Software	System bus data corruption	0.8	0.7	0.2	0.7	Radiate IC in test facility
17	Software	Filesystem corruption	0.99	0.9	0.2	1.071	Radiate IC in test facility
18	Software	Memory leak	0.2	0.3	0.9	0.33	Memory checking programs
19	Software	Memory boundary error	0.3	0.5	0.9	0.6	Automated software testing
20	Software	Failure to handle errors	0.2	0.7	0.1	0.21	Automated software testing

Table 8. Pre-design FMEA risk level.

Table 7 and Table 8 shows the risks of design. The greatest risk are:

- Data corruption
- Physical connection lost
- Invalid sensor data
- Short circuit
- Interference from EM waves

These possible failures will be reduced by the mitigations the design bellow.

5.0 OBC Design

OBC is the core of the Homathko's control system. OBC contains only one microcontroller. The microcontroller is TMS570.

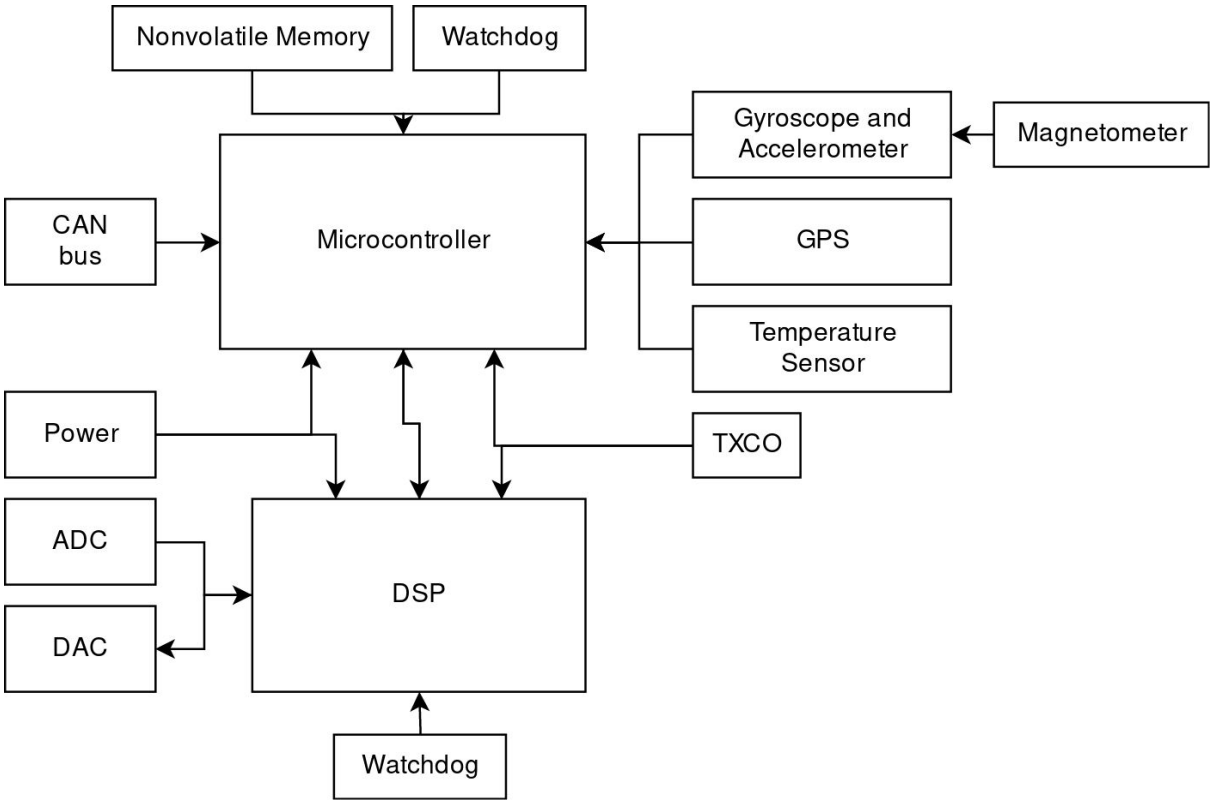

Figure 2. OBC components.

Figure 2 shows the relationship between components in OBC.The ADC and DAC are for the communication system. ADC receives I and Q signals from communications system and sends it to DSP for processing. DSP sends data to DAC. DAC produces the I and Q signals. TXCO is the oscillator for the whole Homathko cube satellite. TXCO provides the clocks for all systems. The microcontroller communicates with DSP for sending and receiving data from ground station. The microcontroller is connected to sensors for the ADCS. Microcontroller is also connected to nonvolatile memory for data storage.

Component	Description	QTY
TMS5703137DZWTQQ1	The main microcontroller.	1
MMC3416xPJ	Magnetometer.	2
ADT7320UCPZ-RL7	Temperature sensor.	1
MPU6050	Gyroscope and accelerometer.	2
SQW-110-01-F-D	GPS connector.	1
Novatel OEM719	GPS.	1
1P1G126QDBVRQ1	IC buffer.	1
TPL5010-Q1	Watchdog timer.	2
MR4A16BCMA35	MRAM.	2
JTAG XDS100	JTAG header.	2
TMS320C5535AZHHA10	Digital signal processor.	1

T200F-010.0M	Oscillator.	1
SN74LVC1G17DSFR	Schmitt trigger.	1
LTC6088HDHC#PBF	General op amp.	1
AD7903BRQZ	Analog to digital converter.	1
CM13032768DZFT	RTC crystal.	1
MCP4921-E/MC	Digital to analog converter.	2
THS4532IPW	Fully differential op amp.	1
TPS7A7200RGTR	GPS LDO.	1
TLV7113333DDSER	MRAM LDO.	1
LP3996SD-2533/NOPB	Dual voltage line LDO.	1
TPS62170DSGR	Switch mode power supply.	1
TPS7A8701RTJR	LDO after switch mode.	1

Table 9. Table of components.

5.1 Hardware Component Design

The purpose and function of each component is presented below.

5.1.1 Magnetometer

MMC3416xPJ was selected as the magnetometer. MMC3416xPJ measures the magnetic flux density in Gauss. MMC3416xPJ has low noise measurement noise . MMC3416xPJ has sensors in the X,Y, and Z axis.
Let a vector point to earth have a magnetic flux density of 0.25 G.

$$\vec{V} = \begin{bmatrix} 0.25 & 0 & 0 \end{bmatrix}$$

The measurement accuracy for each axis is 1.5mG [5].

$$\vec{Z} = \begin{bmatrix} 0.25 + 0.0015 & 0.0015 & 0.0015 \end{bmatrix}$$

Angle between the actual and measured vectors.

$$arccos(\frac{\vec{Z} \cdot \vec{V}}{\left\|\vec{V}\right\| \left\|\vec{Z}\right\|}) = \Theta$$

$$\Theta \approx 0.5^\circ$$

The angle accuracy is around 0.5 degrees. This is similar to the accuracy in the datasheet of 1 degree accuracy.

MMC3416xPJ will be used to detect detect the earth's magnetic field. Magnetic flux density data would allow the OBC to determine the attitude of Homathko. The range of the MMC3416xPJ is close to the variance in the earth’s magnetic flux density.

MMC3416xPJ uses I2C to communicate with MPU6050. The Kalman filter does not need that many updates. The frequency of polling would be 13 Hz. 36 bits is required for a one vector. 100kHz speed for I2C is more than enough.

The minimum pull up resistor value for the sensor.

$$Rp(min) = \frac{V_{CC} - V_{OL}}{I_{OL}} = \frac{3.3V - 0.4V}{3mA} \approx 1k\Omega$$

At 100 kHz with 100pF of load capacitance.

$$Rp(max) = \frac{t_r}{0.8473C_b} = \frac{1 \times 10^{-6}s}{100pF} \approx 12k\Omega$$

10 kOhms was chosen as the pull up resistor to conserve power.

Two MMC3416xPJs will be placed on the OBC for redundancy.

5.1.2 Gyroscope and Accelerometer

MPU6050 was chosen for the gyroscope and accelerometer. MPU6050 interfaces directly with MMC3416xPJ using I2C. MPU6050 acts as I2C switch to connect and disconnect MMC3416xPJ.

MPU6050 measures acceleration and angular velocity when Homathko is spinning. The maximum range of the gyroscope is 2000 degree/s [6]. This is sufficient to detumble Homathko. At rest, the sensor does not give any information about the attitude.

The MPU6050 has a FIFO that buffers data from MMC3416xPJ and internal sensors. If the buffer is full then MPU6050 will raise the INT pin for an interrupt. The TMS570 will read the buffer from MPU6050. MPU6050 also has a temperature sensor to compensate temperature fluctuations.

The detumble algorithm requires gyroscope data at 1 kHz to detumble Homathko. Therefore the I2C speed must be 400kHz.

$$Rp(min) = \frac{V_{CC} - V_{OL}}{I_{OL}} = \frac{3.3V - 0.4V}{3mA} \approx 1k\Omega$$

At 400 kHz with 200pF of load capacitance.

$$Rp(max) = \frac{t_r}{0.8473C_b} = \frac{300 \times 10^{-9}s}{200pF} \approx 1.77k\Omega$$

The closest value would be 1kOhm for pullup resistor.

Two MPU6050s will be placed on the OBC for redundancy. Each one has a separate I2C address selected by the I2C address pin. The unused MPU6050 will sleep to conserve power.

5.1.3 Temperature Sensor

ADT7320UCPZ-RL7 was chosen for the temperature sensor. ADT7320UCPZ-RL7 will be used as a backup temperature sensor. ADT7320UCPZ-RL7 warns the TMS570 if the OBC's temperature get too high. ADT7320UCPZ-RL7 has interrupt lines directly to TMS570. ADT7320UCPZ-RL7 uses SPI to communicate with TMS570. The update frequency to the TMS570 is 1 Hz.

5.1.4 GPS

Novatel OEM719 was chosen as the GPS. Novatel OEM719 provides the location of Homathko within 10 meters [7]. Novatel OEM719 communicates with TMS570 using a serial interface. The serial interface allows TMS570 to configure settings in the Novatel OEM719. Novatel OEM719 will be connected to the OBC using a connector and a ribbon cable. The update rate will be 100 Hz.

Novatel OEM719 also provides a GPS PPS. GPS PPS is generated once every second. The rising edge of one GPS PPS to another rising edge GPS PPS is approximately one second. GPS PPS's time accuracy is 100 ns. GPS PPS has a buffer to increase the current output of the signal. GPS PPS signal is used to synchronize the clocks on all the MCUs on Homathko. It also helps with time synchronization with ground station.

Novatel OEM719 uses 1 W when active. This consumes too much power for continuous use. Therefore, Novatel OEM719 will only be turned on every 4 orbits to conserve energy. The GPS turn on procedure is always the same. Firstly, TMS570 will turn on the Novatel OEM719. Secondly, Novatel OEM719 will give Homathko's position to the TMS570. Thirdly, all the clocks on Homathko will be synced. Fourthly, internal orbital simulations on Homathko will use Novatel OEM719's position as a starting position to guess the orbit path. Lastly, Novatel OEM719 will be powered down after a few minutes.

5.1.5 CAN bus transceiver

SN65HVD234DR was choosen as the CAN bus transceiver. SN65HVD234DR accepts CMOS logic from TMS570 and converts it to CAN bus signals. ISO 11898-2 specifies the CAN bus requirements. CAN bus operates at 500 kbps with 120 ohm termination.

There are two SN65HVD234DR for redundancy. Two CAN buses gives a higher throughput for data transfer than one CAN bus. SN65HVD234DR has a low power mode that allow the IC to conserve energy. SN65HVD234DR also has the ability to only listen for message while in low power mode.

5.1.6 Watchdog Timer

TPL5010-Q1 was chosen as the watchdog timer. Two TPL5010-Q1s are placed on the OBC. One for TMS570 and one for TMS320C5535. TPL5010-Q1s are used to reset the ICs when they fail to respond. TPL5010-Q1 will trigger a cold boot when it didn't receive a DONE

signal from TMS570. TMS570 needs a few minutes to boot up from an off state. Therefore, TPL5010-Q1's timer is set for 10 mins.

The external resistance determines the time interval [8].

$$R_{EXT} = 100\frac{-b + \sqrt{b^2 - 4a(c - 100T)}}{2a}$$

10 mins is 600 s. T will be 600.

$$100 < T \leq 1000$$
$$a = 0.2617$$
$$b = -56.2407$$
$$c = 5957.7934$$
$$R_{EXT} = 100\frac{56.2407 + \sqrt{56.2407^2 - 4 \cdot 0.2617(5957.7934 - 100 \cdot 600)}}{2 \cdot 0.2617}$$
$$R_{EXT} = 57.441k\Omega$$

REXT will be around 57.441 kohms. A 107 kohm resistor and 124 kohm will be used in parallel to make 57.437 kohms. The error will be less than one second.

5.1.7 Nonvolatile Memory

MR4A16BCMA35 was chosen as as the nonvolatile memory storage. MR4A16BCMA35 stores the all the program's memory in Homathko. MR4A16BCMA35 uses MRAM technology. MRAM is magnetoresistive random access memory where the bits are stored in magnetic fields. There are two MR4A16BCMA35s on OBC. Each MR4A16BCMA35 stores 2MB of data.

MRAM uses magnetic tunnel junction to store information. Magnetic tunnel junction consists of two ferromagnets and one insulator. The two ferromagnets are separated by the insulator. One ferromagnet is permanent. The other is a free layer. A large surge of current is need to change the orientation of the free layer. Orientation determines the value of the bit stored. Magnetic tunnel junction makes MRAM radiation resistant to cosmic rays.

Cosmic rays consists mostly of heavy ion bombardments. Heavy ions may change the value of charge pumps. On the other hand, magnetic tunnel junction resists changes in the magnetic field. It is improbable for heavy ions to generate a large enough current to change the magnetic field. Therefore, the Single Event Upset(SEU) and Single Event Latchup(SEL) damage thresholds are large. If data is stored using ferromagnetic elements then data retention period depends on the lifetime of the ferromagnetic elements. Ferromagnetic elements will preserve their magnetic fields for at least 20 years.

Parameter	Limits
Data retention	>20 Years
TID max	40 krad

SEU	>100 MeV-cm^2 /mg
SEL	>84 MeV-cm^2 /mg
Access time	35 ns
ECC	7 bits parity per 64 bits
H field tolerance	8000 A/m

Table 10. Properties of Everspin MRAM [9] [10] [11].

MR4A16BCMA35 has 20 years of data retention as stated in Table 10. Mission duration is two years.

Calculation of TID from dose rate. MRAM area is 1 cm^2.

$$2794.8086\frac{MeV \cdot cm^2}{g \cdot s} \times \frac{1}{1cm^2} = 2794.8086\frac{MeV}{g \cdot s}$$

$$2794.8086\frac{MeV}{g \cdot s} \times \frac{1000g}{1kg} \times \frac{1.60218 \times 10^{-13}J}{1MeV}$$

$$= 4.477786 \times 10^{-7}\frac{J}{kg \cdot s}$$

$$4.477786 \times 10^{-7}\frac{J}{kg \cdot s} \times \frac{100rad \cdot kg}{J \cdot s} = 4.477786 \times 10^{-5}\frac{rad}{s}$$

After two years in seconds is 6.307×10^7 s. The TID in LEO.

$$TID_{LEO} = 4.477786 \times 10^{-5}\frac{rad}{s} \times 6.307 \times 10^{7}s = 2824rad$$

$$TID_{max} > TID_{LEO}$$

$$40000rad > 2824rad$$

TID for two years in LEO is 2824 rad. The maximum TID of MR4A16BCMA35 can withstand is 40 krad. The maximum TID for MR4A16BCMA35 is greater than TID of LEO. Therefore, MR4A16BCMA35 will survive the mission.

$$SEU_{thres} = 100\frac{MeVcm^2}{mg} = 100000\frac{MeVcm^2}{g}$$

$$SEU = 2794\frac{MeVcm^2}{g}$$

$$SEU_{thres} > SEU$$

The MR4A16BCMA35 SEU threshold is far greater than SEU. Therefore SEUs will have little effect on MR4A16BCMA35.

The number of upsets or bit errors in 4MB in entire mission duration [10]. Two years is 730 days.

$$p = 1 \times 10^{-10} \frac{upsets}{bit \cdot day} \times 730\ days \times 1\ bit$$

$$p = 7.3 \times 10^{-8} upsets$$

$$N = 3.2 \times 10^{7}$$

$$f = \binom{N}{k} p^{k} (1-p)^{N-k}$$

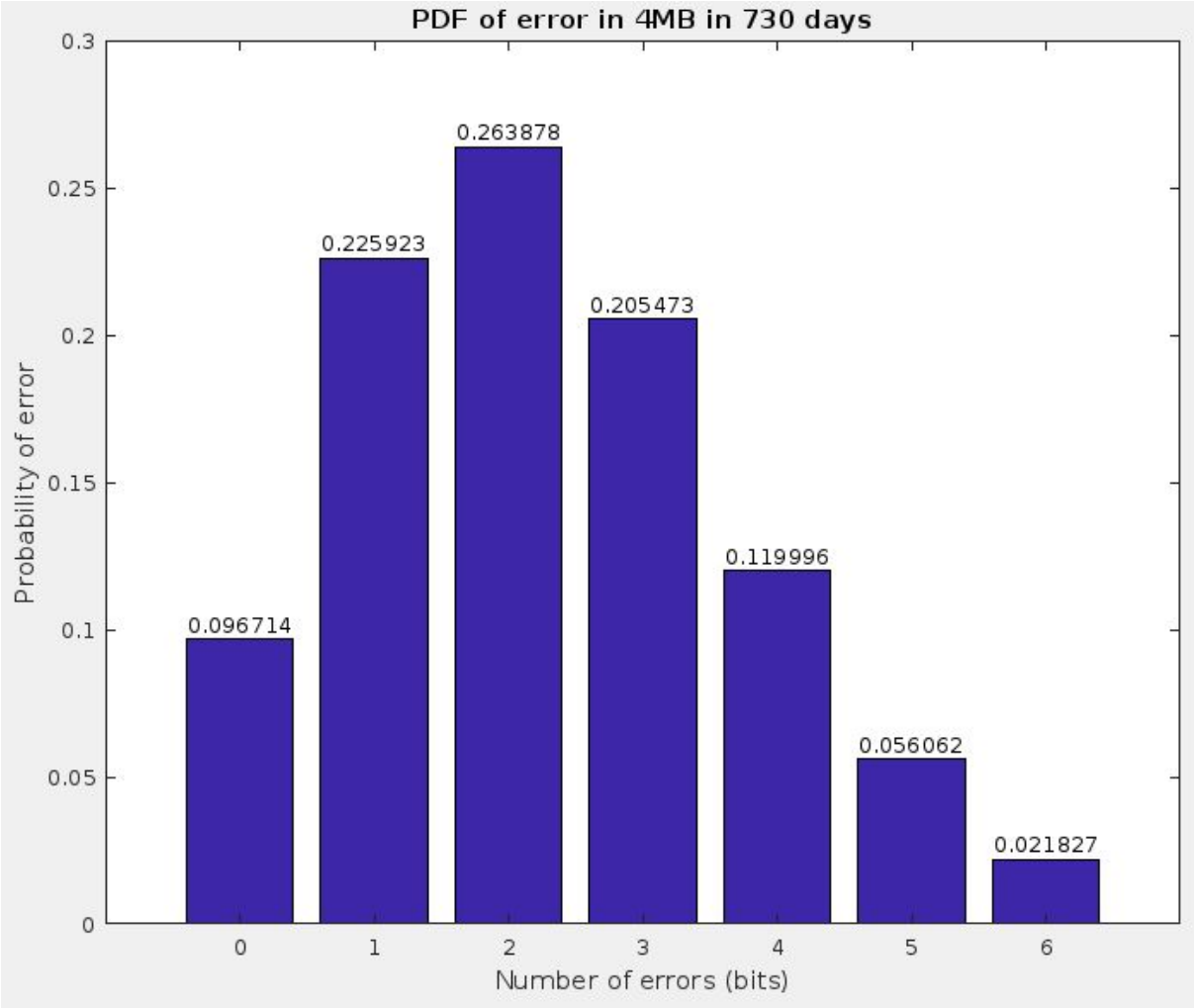

Figure 3. PDF of MRAM 4MB in 730 days.

Figure 3 shows the probability of a bit errors is very small. There is a mean of 2 bit errors over 730 days. This is without hardware ECC in MR4A16BCMA35.

In worst case scenario, MR4A16BCMA35 is powered off for 1 day. MR4A16BCMA35 uses 7 bits of parity for every 64 bits. MR4A16BCMA35's ECC recovers 1 bit for every 71 bits.

$$p = 1 \times 10^{-10} \frac{upsets}{bit \cdot day} \times 1\ day \times 1\ bits$$

$$p = 1 \times 10^{-10} upsets$$

$$N = 71$$

$$f = \binom{N}{k} p^k (1-p)^{N-k}$$

$$f(k > 1) = P_{71bits}$$

The probability of ECC not correcting bit error in 71 bits. $P_{71bits} = 6.6613 \times 10^{-16}$
There are 5E5 segments of 71 bits to make up the total memory capacity.

$$N = 5 \times 10^5$$

$$f = \binom{N}{k} p^k (1-p)^{N-k}$$

$$f(k > 0) = P_{4MB}$$

The probability of error ECC not correcting bit errors in 4MB in 1 day.

$$P_{4MB} = 3.331 \times 10^{-10}$$

The probability of error when MR4A16BCMA35 is turned off is very low. All other scenarios have significantly lower probability of error.

There are two MR4A16BCMA35s on OBC. One MR4A16BCMA35 uses EMIF on TMS570. This MR4A16BCMA35 is the primary data storage for TMS570. This MR4A16BCMA35 can not be turned off. TMS570 could write 16 bits of data with a 20 bit address using EMIF. TMS570 uses enable, write, low byte, high byte, and output pins as control for MR4A16BCMA35.

The other MR4A16BCMA35 uses GPIO pins on TMS570. This method allows both MR4A16BCMA35s to operate in parallel. However, the power on this MR4A16BCMA35 could be turned off. Then the GPIO pins would be grounded to prevent ghost power draw.

5.1.8 TXCO

T200F-010.0M is the main oscillator for Homathko. T200F-010.0M is a temperature controlled oscillator [12]. Therefore the frequency drift is very little. T200F-010.0M's frequency will not vary over large temperature swings. T200F-010.0M generates a square wave at 10MHz and the output is sent to a schmitt trigger. Schmitt trigger reduces noise in the square wave. Many clocks will use this oscillator as a reference. TMS570 uses this clock for the internal PLLs and system clocks. TMS320C5535 also uses the clock for internal PLLs.

5.1.9 ADC

AD7903BRQZ is the ADC for TMS320C5535. AD7903BRQZ samples the differential signals from the communications system. AD7903BRQZ has a sample rate of 1MSPS with 16 bits of resolution [13]. 1MSPS meets the specification of 198 kHz sample rate. 16 bits of resolution is sufficient for a QPSK system.

AD7903BRQZ has one SPI interface for each I and Q signal. I+ and I- are sampled to create I. Q+ and Q- are sampled to create Q. The signal is feed into TMS320C5535 using MISO. The signal is synced with SPI clocks. Both SPI clocks are synced to reduce errors in delay. SPI clock is 1 MHz.

LTC6088HDHC#PBF is a general operational amplifier. LTC6088HDHC#PBF is used as an input buffer for AD7903BRQZ. There are four amplifiers on one package. The slew rate of LTC6088HDHC#PBF is 7.2V/μs. This sufficient for the sample rate.

5.1.10 DAC

MCP4921-E/MC is used as the DAC for the communications system. One DAC is for I signal and one DAC is for Q signal. DAC resolution is 12 bits at 5.5 µs settling time [14]. The settling time of DAC is sufficient for a 198 kHz rate.

MCP4921-E/MC has one SPI interface for each I and Q signal. The layout is similar to the ADC. SPI clock is 1 MHz. Each I and Q signal is connected to a fully differential amplifier. The THS4532IPW is the differential amplifier. THS4532IPW is used for a conversion from a single ended signal to a two ended signal. The THS4532IPW has a gain of one and a slew rate of 200 V/µs.

5.1.11 DSP

TMS320C5535 is the DSP for the communications system. TMS320C5535 will run at 100MHz with an input clock of 10 MHz [15]. A separate RTC is also used to keep time. GPS PPS is connected to the DSP to sync the times on the DSP. A watchdog timer is connected to the reset pin of TMS320C5535. The watchdog timer resets TMS320C5535 when it does not respond to wake up signals. There is an interrupt line from TMS320C5535 to TMS570. If TMS320C5535 receives data from the communications system then it will send an interrupt signal to the TMS570. This will inform the TMS570 of incoming communications data.

TMS320C5535 communicates with TMS570 using UART. TMS320C5535 will boot from UART by receiving program data from TMS570. TMS570 also scrubs the memory on the TMS320C5535 using UART. UART allows TMS320C5535 to transmit and receive communications data to and from TMS570.

TMS320C5535 uses two SPI interfaces to communicate with two DACs and two ADCs. TMS320C5535 also controls the enables for amplifiers. JTAG is used to program and debug the TMS320C5535. XDS100 will connect to the JTAG connector on the OBC. USB port, I2C port and built-in ADC on TMS320C5535 are disabled to conserve energy.

5.1.12 Microcontroller

TMS570 is the main microcontroller for the OBC. TMS570 controls everything on the Homathko. TMS570 uses a dual lockstep CPU [16]. THe dual CPUs are identical. For every instruction, dual CPUs on the TMS570 executes the exact same instruction. If the same instruction produces different results then both CPUs will re-execute the instruction again. Dual lockstep CPUs protects against SEUs.

TMS570 also has a self testing capabilities to detect errors in hardware. At bootup, the TMS570 will check all of the builtin modules. TMS570 also has eFuses with parity bits. eFuses uses fuse resistances to store data. eFuses protects against radiation as radiation can not damage resistors. TMS570 survives up to 5 krad of TID. TMS570 underwent heavy

ion bombardment up to 5 krad at the Canadian Space Agency test facilities. Therefore, TMS570 will survive 2824 krad of radiation during the whole mission.

H is the number of hours. P is the probability of error in a bit. N is the number of bits. F is the probability of bit errors [4].

$$H = 24$$

$$p = 1 - e^{-H*9.0813\times10^{-6}}$$

$$N = 72$$

$$f = \binom{N}{k} p^k (1-p)^{N-k}$$

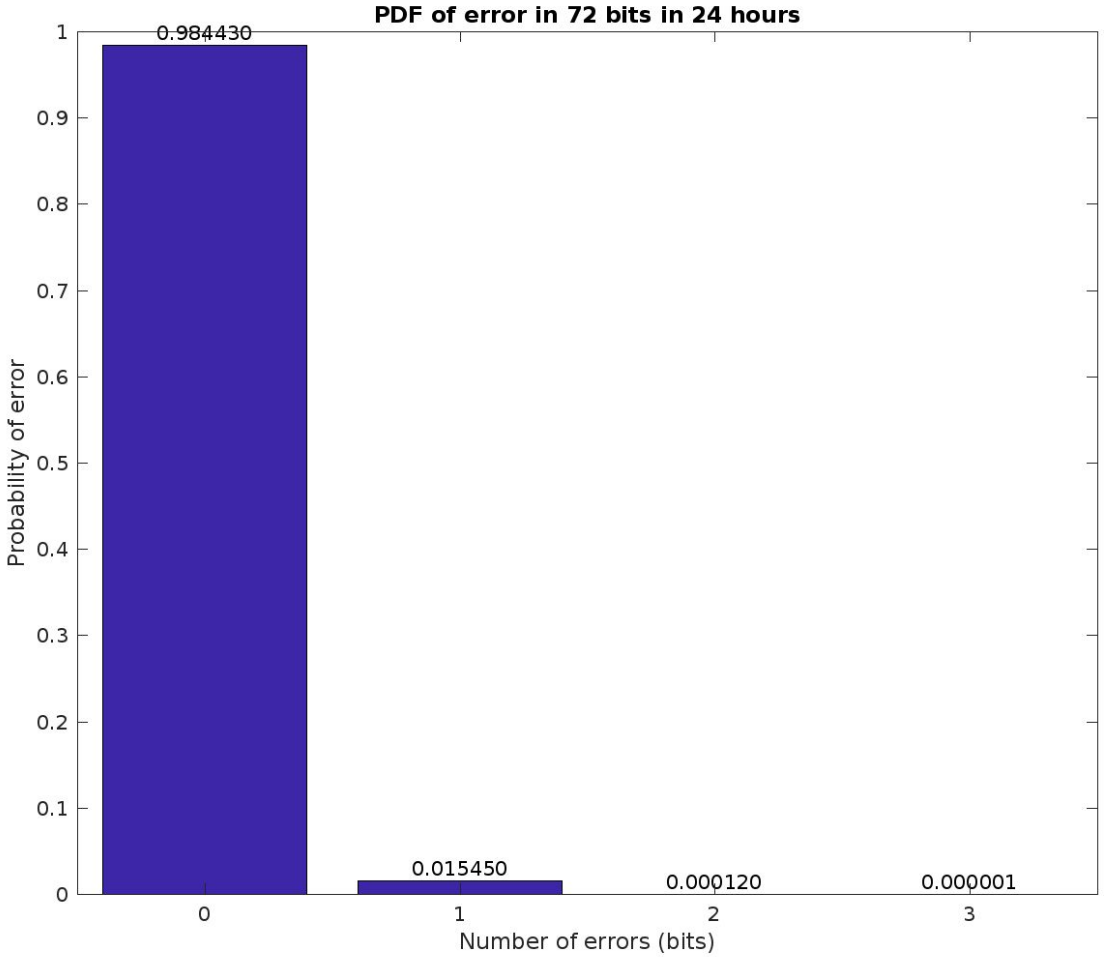

Figure 5. PDF of error in 72 bits.

TMS570 has hardware ECC with 64/72 bit hamming. TMS570 sitting in the rocket for 24 hours at solar maximum is the worst case scenario. Calculations only assumes there is only 2mm Al shielding. Calculations do not take in account the shielding from the rocket. TMS570's ECC can correct 1 bit for every 72 bits in the internal memory. Figure 4 shows the probability of TMS570's ECC not correcting the bit error is 0.00012 in 72 bits.

$$f(k > 1) = P_{72bits}$$
$$P_{72bits} = 0.00012$$

The bootloader will be less than 5000 bits. There is 70 segments of 72 bits for the bootloader.

$$N = 70$$
$$f = \binom{N}{k} p^k (1-p)^{N-k}$$

The probability of ECC not correcting bit errors in 5000 bits.

$$f(k > 0) = P_{5000bits}$$
$$P_{5000bits} = 0.008365$$

The probability of error is 0.008365 for the bootloader.

Calculations for 3MB in 24 hours in TMS570.

$$H = 24$$

$$p = 1 - e^{-H * 9.0813 \times 10^{-6}}$$

$$N = 2.4 \times 10^{7}$$

$$f = \binom{N}{k} p^{k} (1-p)^{N-k}$$

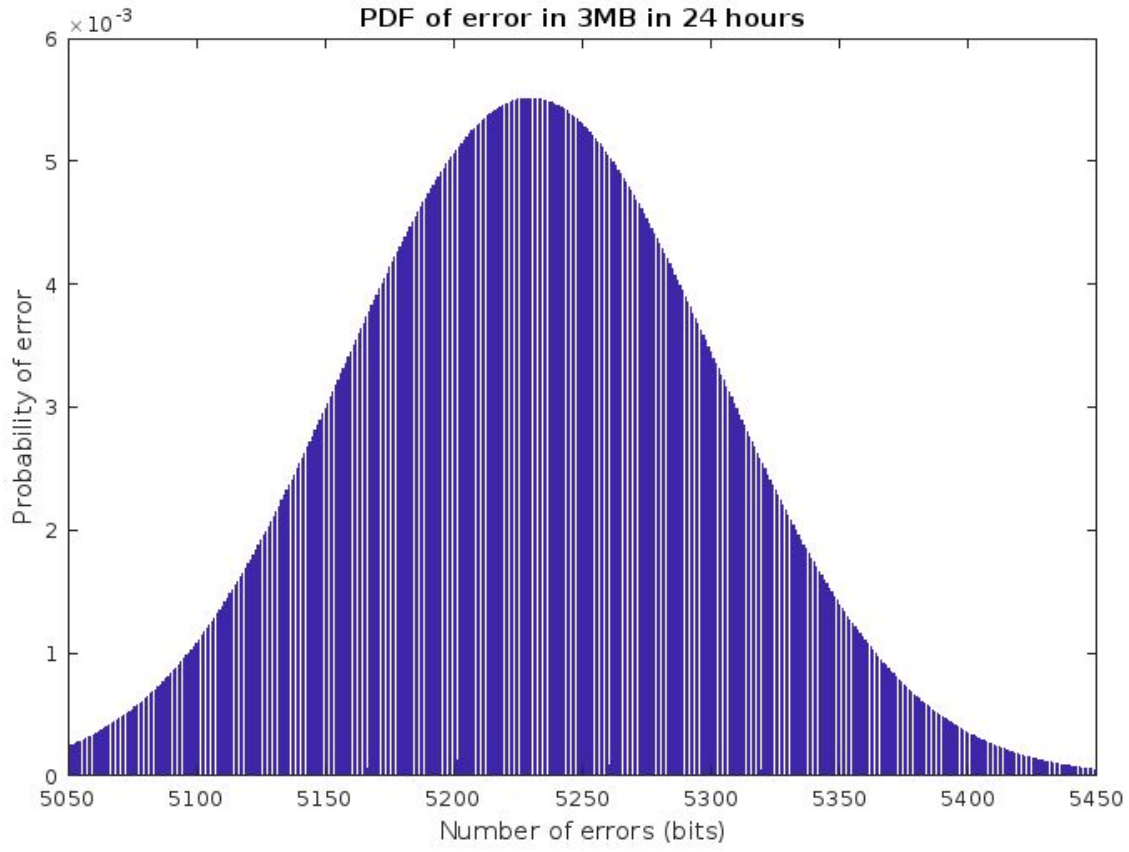

Figure 5. PDF of error in 3MB in 5 hours.

Figure 5 shows the mean of bit errors is 5225 bits. There is a standard deviation of 50 bit errors. If the memory scrubbing time for TMS570 is 24 hours then the bit errors are very large. A file system design is needed to reduce the bit errors.

T200F-010.0M provides the clock for the TMS570. The system clock of TMS570 is a multiple of T200F-010.0M's 10 MHz clock. The built in ADC module for TMS570 is disabled. All pins in the ADC module are grounded.

Watchdog timer is attached to TMS570's reset pin. If TMS570 fails to send done signal then watchdog timer would force a cold reboot. TMS570 uses JTAG for programming and debugging.

TMS570 is connected to one MR4A16BCMA35 using EMIF. EMIF consists of controls pin, address pins, and data pins. The control pins enable MR4A16BCMA35 at the start of a operation. Control pins also sets the read and write modes of MR4A16BCMA35. TMS570 uses 20 bit address for MR4A16BCMA35. Data pins are 16 bits and connect to EMIF. The other MR4A16BCMA35 is connected using GPIO pins instead of EMIF.

TMS570 has a pins connected directly to the edge connector. Enable pins on the edge connector are used to enable certain subsystems in Homathko. Reset pins on the edge connector are used to force STM32s into CAN bootloader mode. The rest of edge connector are unassigned GPIO.

5.2 Software Design

5.2.1 Software Stack

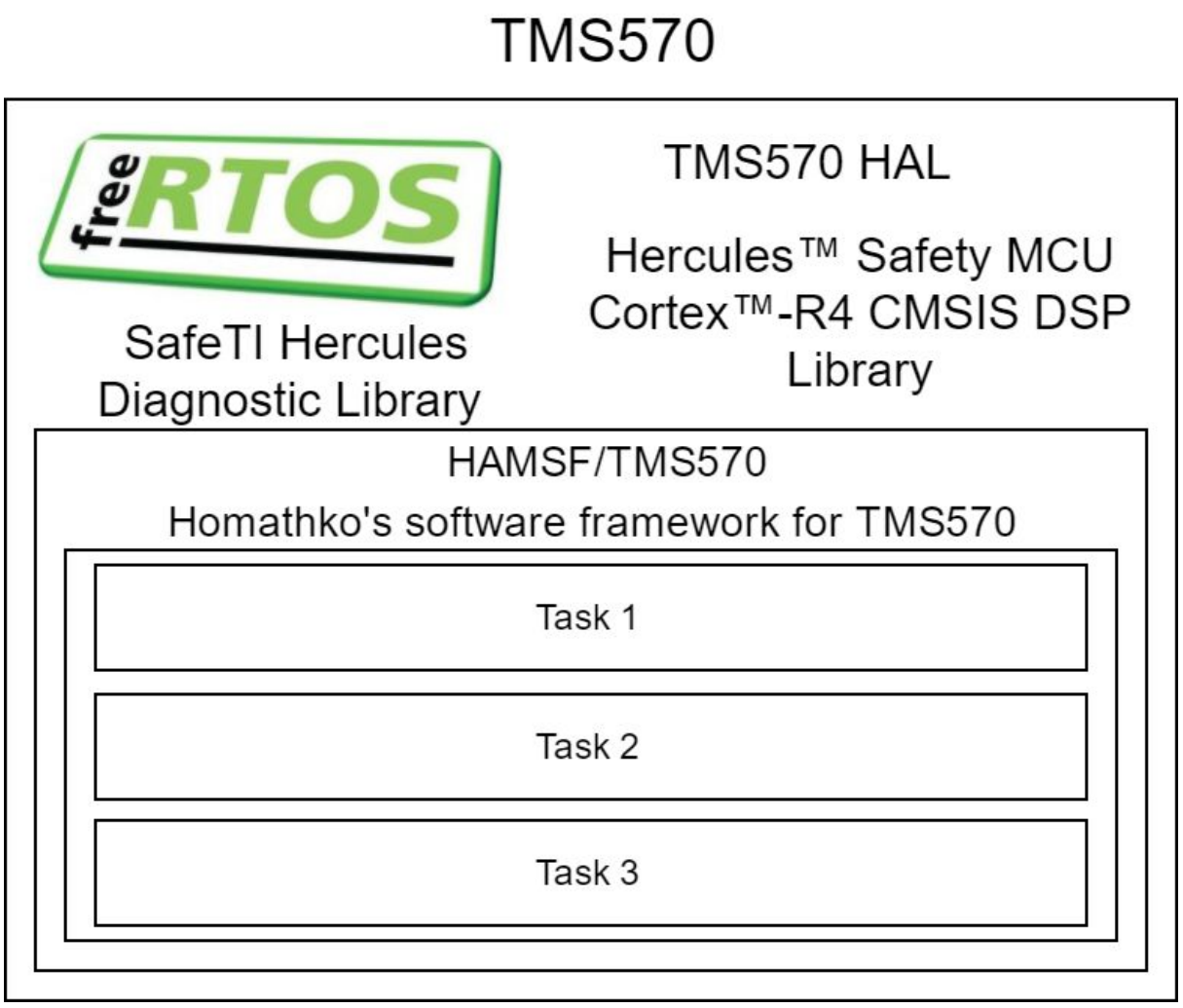

Figure 6. Software stack of TMS570.

Figures 6 shows the software stack of TMS570. TMS570's software will use FreeRTOS for the operating system. TMS570 HAL and SafeTI Hercules Diagnostic Library will be used to develop tasks and other frameworks. A custom library (HAMSF/TMS570) for Project Homathko will be created to share code between subsystems. Tasks will use the the libraries listed above. The DSP library will be used for the ADCS. ADCS design document will provide specifics.

FreeRTOS was chosen because it was lightweight and simple. FreeRTOS will run on all the MCUs on the Homathko. FreeRTOS is also modular and extendable. Moreover, FreeRTOS is a real time operating system for embedded systems. FreeRTOS is able to preemptively execute tasks. Preemption increases the timing accuracy of tasks and interrupts. FreeRTOS will run on directly MR4A16BCMA35 instead of the internal memory of TMS570.

FreeRTOS will use a round robin scheduler with interrupts. Round robin scheduler is optimal for controlling tasks on FreeRTOS. Interrupts will be used to warn the microcontroller of high priority tasks. For example, if the battery system is running out of energy then the microcontroller will be warned. After the interrupt fires, the microcontroller will turn off certain systems to conserve energy.

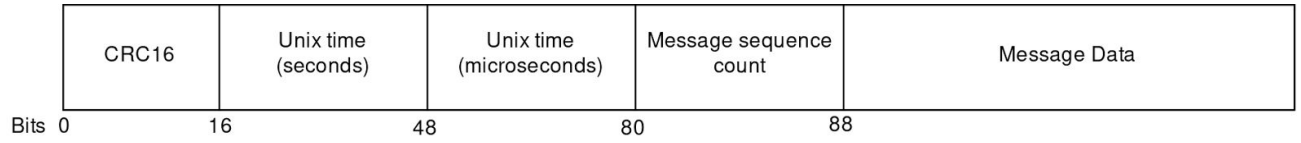

Figure 7. Queue system for system bus.

The default queue system in FreeRTOS is not very robust. Therefore, default queue system will be modified. The new queue system will be cyclic. This is to prevent buffer overflows. CRC16 will be used to check the integrity of the message as shown in Figure 7. CRC16 detects up to 16 bit errors in the message. There is a timestamp and message sequence count. Timestamps are used for the Kalman filter. Message sequence count is used to detect missing messages on system bus.

5.2.2 File System Design

File system will be used to store all of Homathko's program data. File system will be present on all MCUs on Homathko.

Program name	Size (bytes)	Priority	Storage Location
TMS570LS3137PGE SPI bootloader	500	HIGH	Flash,MRAM
Memory scrubbing + Self diagnostics	300000	HIGH	Flash,MRAM
FreeRTOS kernel	10000	MEDIUM	MRAM
TMS570 HAL	500000	MEDIUM	MRAM
HAMSF	10000	MEDIUM	MRAM
Power control	10000	MEDIUM	MRAM

Timesync	2000	MEDIUM	MRAM
Watchdog	2000	MEDIUM	MRAM
CAN bus	10000	MEDIUM	MRAM
Inflight reprogramming	10000	MEDIUM	MRAM
Communications	200000	MEDIUM	MRAM
Payload control	5000	MEDIUM	MRAM
TMS320C5535 program	600000	MEDIUM	MRAM
STM32 Payload	50000	MEDIUM	MRAM
STM32 Solar panel	20000	MEDIUM	MRAM
STM32 ADCS	40000	MEDIUM	MRAM
STM32 Power monitor	20000	MEDIUM	MRAM
ADCS	1.80E+06	LOW	MRAM
Logging	200000	LOW	MRAM
Total	3789500		

Table 11. Estimate of program data size.

Table 11 shows all of the programs and program data size. The total size of program data is around 3.8 MB. A file system is required to protect that data.

File system will use similar specifications determined by proposed FTRFS paper [17]. FTRFS uses Reed Solomon codes to protect the file system. Reed Solomon codes will protect the file system against bit errors by using parity bits. The FTRFS mentions using 68 byte parity for a 128 byte block in the super block. Super blocks contain critical information about the filesystem. Super blocks details the size of the file system and location of Inode blocks. Multiple super blocks will be used for redundancy. Inode block uses 68 byte parity for a 160 byte block. Bitmap uses 688 byte parity for 1773 byte block. In addition, CRC32 will be used to check the integrity of a file.

The number of parity bits for the data block is dynamic. It changes depending on the type of device or the amount of radiation. File system monitors radiation dose rate. If radiation increases then parity bits will increases at run time.

Reed Solomon is a subclass of BCH codes. The actual file system will use BCH codes instead of Reed Solomon codes. BCH codes are more useful for single bit errors. For every mt parity bits in BCH code, it can recover up to t bits. Assume a 4095 bit block with a scrubbing period of 6 hours in TMS570's memory.

$$M = 12$$
$$N = 2^M - 1 = 4095$$
$$H = 6$$

$$p = 1 - e^{-H*9.0813\times10^{-6}}$$

$$f = \binom{N}{k} p^k (1-p)^{N-k}$$

$$P_{4096bits} = f(k > t)$$

$$S = \frac{2.4 \times 10^7}{4096} = 5.859 \times 10^3$$

$$g = \binom{S}{x} P_{4096bits}^X (1 - P_{4096bits})^{S-X}$$

$$P_{error} = g(X > 0)$$

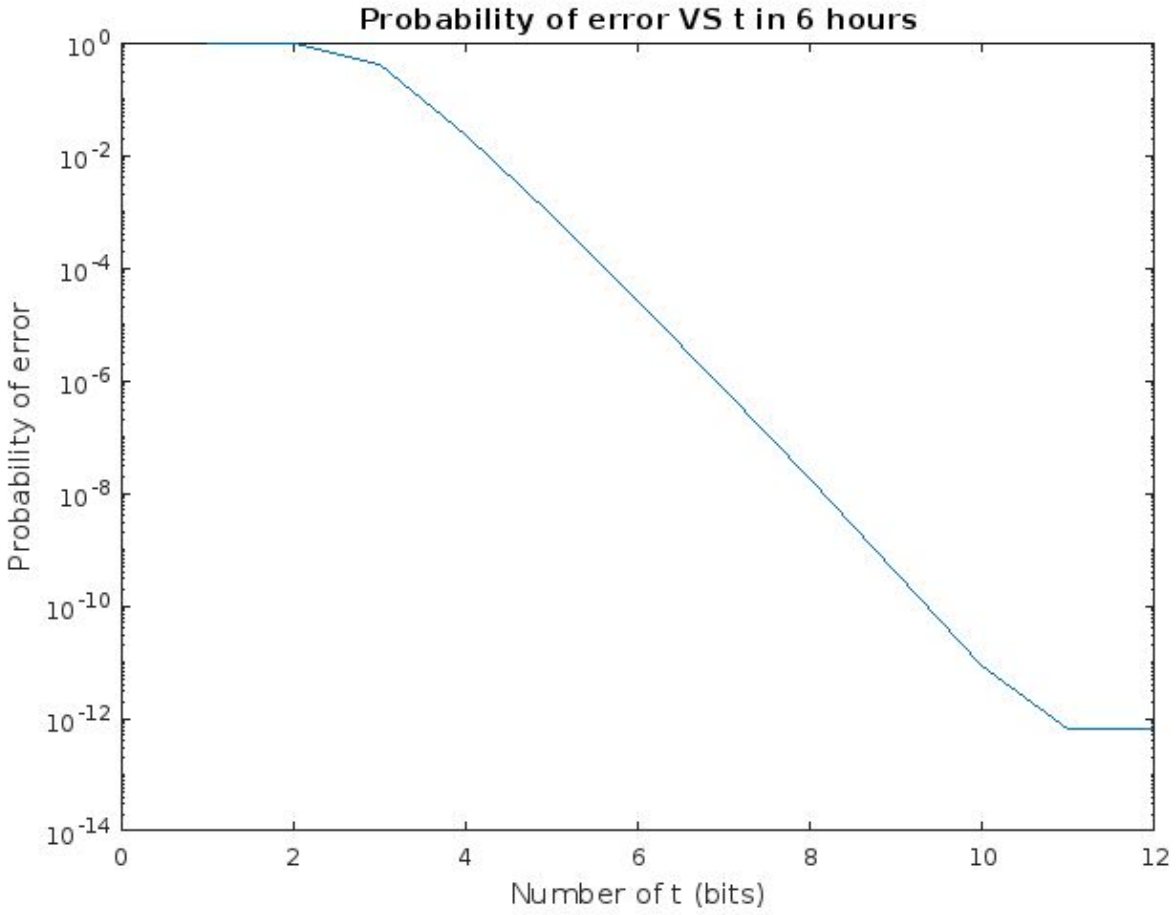

Figure 8. Probability of error vs number of t bits. TMS570.

For a low probability of error select t = 9 bits as shown in Figure 8.

$$P_{error} \leq 10^{-10}$$

$$t = 9$$

$$M * t = 108$$

There are 108 parity bits in a 4095 bit block. It corrects up to 9 bit errors. It will be sufficient for TMS570's 3MB memory.

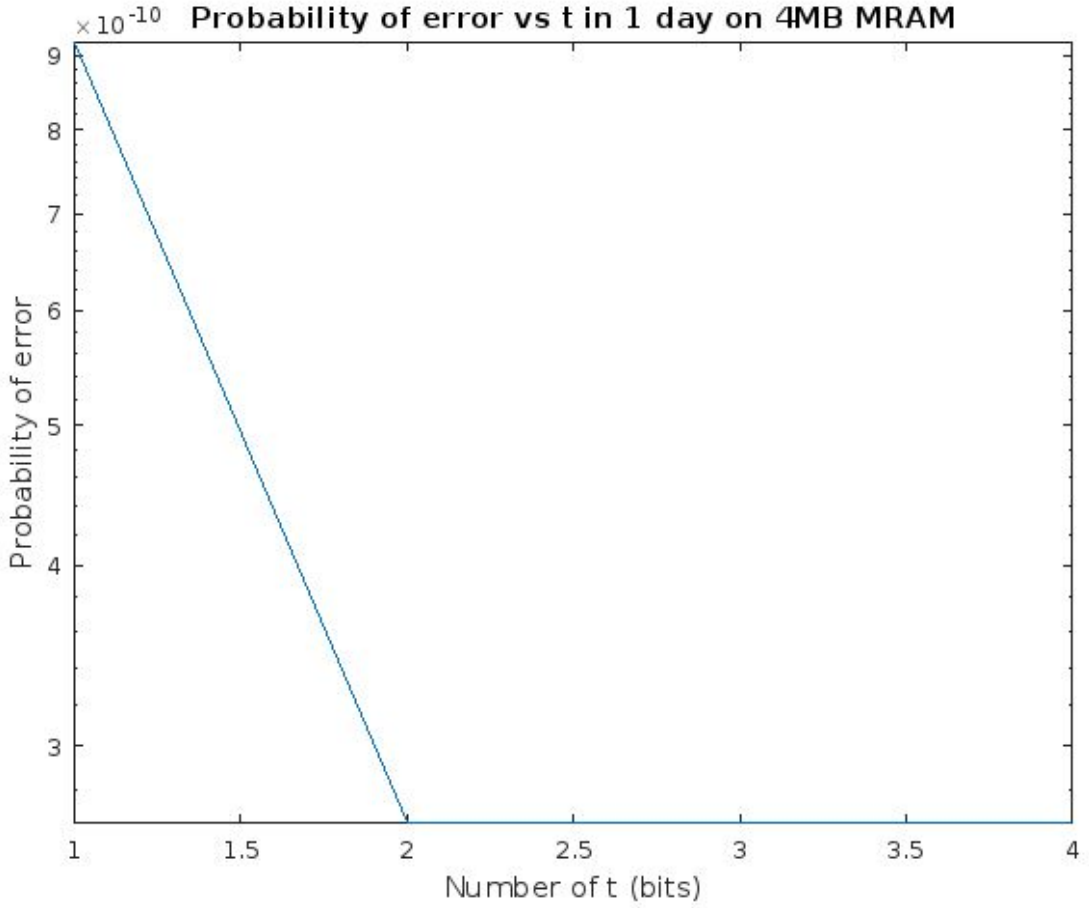

Figure 9. Probability of error vs number of t bits. MRAM.

The same calculations were done for MRAM. Figure 9 shows when t = 2 then probability of error is low.

$$t = 2$$

$$M * t = 24$$

There is 24 parity bits in a 4095 bit block. It corrects up to 2 bit errors. MRAM requires less parity bits because it is more resilient to radiation. MRAM's scrubbing period is one day.

5.2.3 Communications Stack

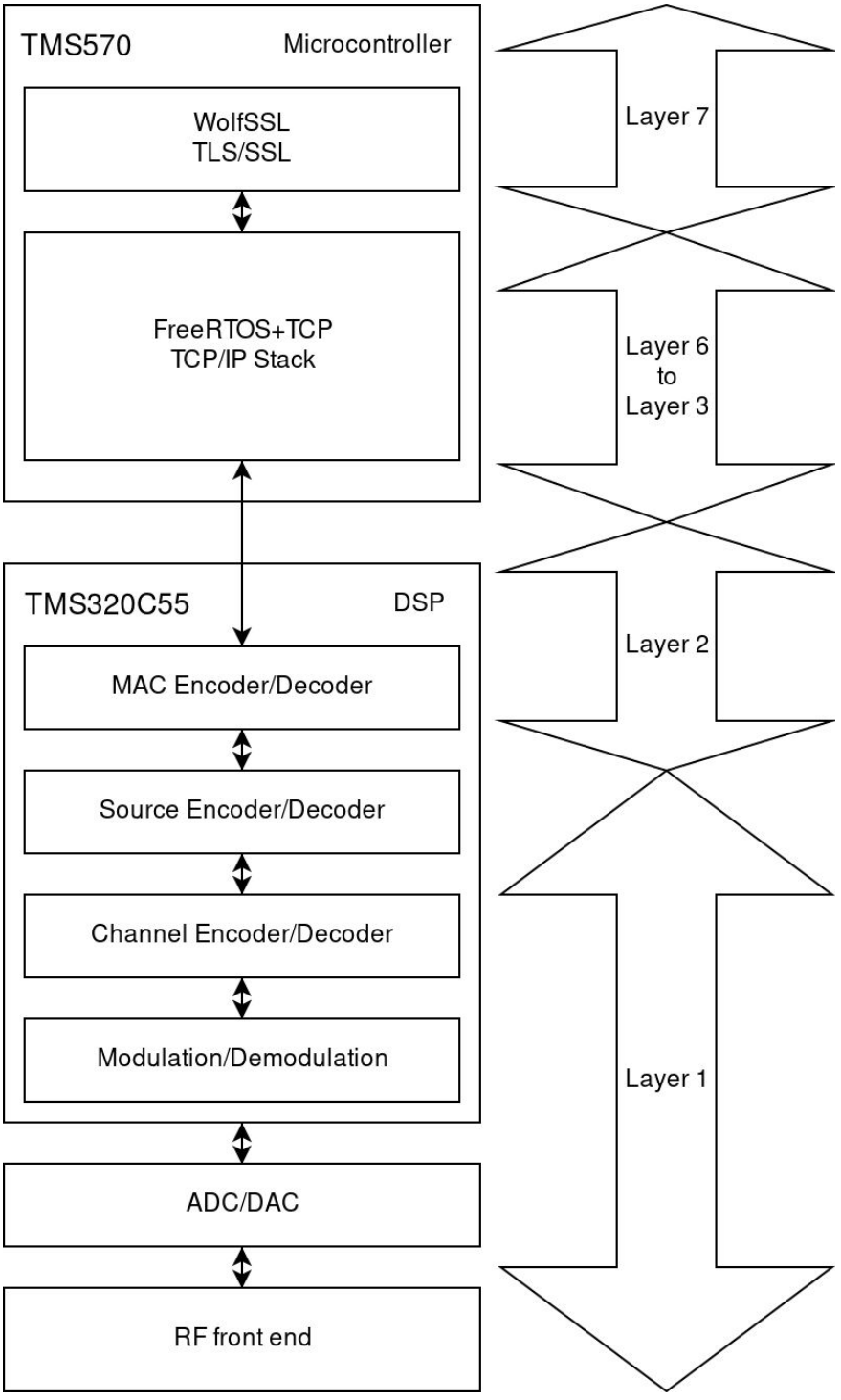

Figure 10. Communications stack in software.

Figure 10 presents the communications system. RF front end and TMS320C5535 will implement layer 1. TMS320C5535 will also implement layer 2. Layer 2 implements MAC header framing. TMS570's software will implement the rest of the layers. Layer 3 will implement the IP using FreeRTOS+TCP. Layer 4 will implement TCP using FreeRTOS+TCP. FreeRTOS kernel will implement layer 5 to layer 6. Finally, WolfSSL will

implement layer 7. WolfSSL will use TLS 1.2 for encrypting and decrypting remote commands.

5.2.4 Finite State Machine Design

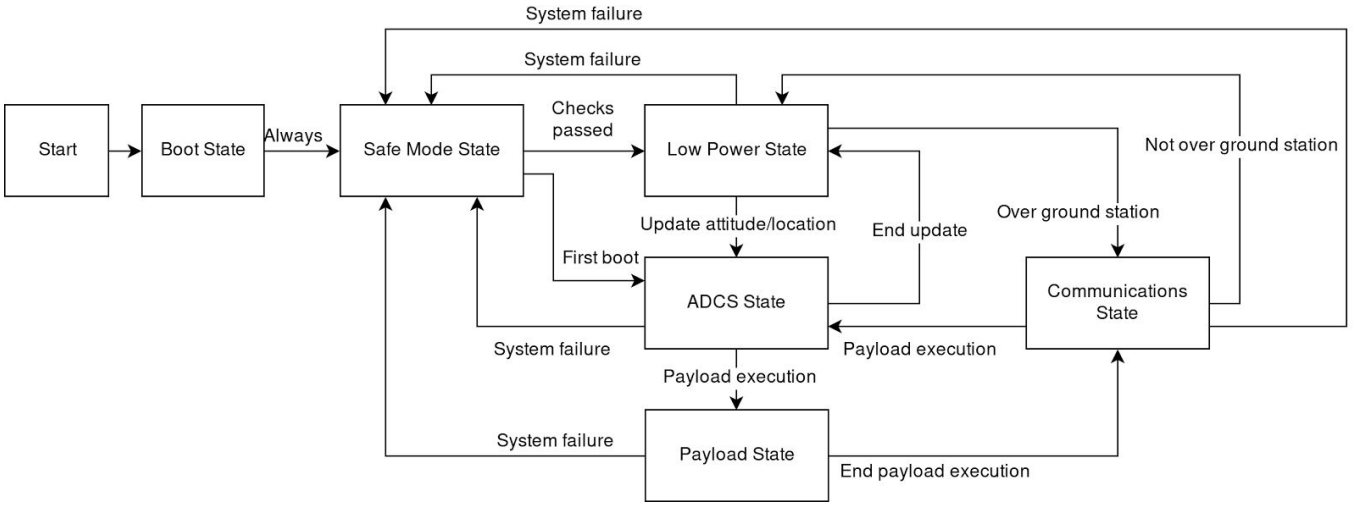

Figure 11. Finite state machine for OBC.

The software starts with Boot State every time as shown in Figure 11. Boot State will load necessary software for Safe Mode State. After booting, the software will transition to Safe Mode State. Safe Mode State perform system checks. On first boot, ADCS State will be turned on for GPS contact. Low Power State is the default state of Homatko. OBC will be in Low Power State for 99% of the time.

ADCS State will be responsible for updating location and attitude of the satellite. It turns on GPS, magnetometers, gyroscope, and thermopiles. ADCS State will also be activated just before Payload State to get accurate location.

If Homathko is over the ground station then it will transition into Communications State. Communications State will try to contact the ground station. If Homathko receives a command to execute the payload then Homathko will transition into ADCS State at the specified time. ADCS State will orient Homathko just before Payload State. After that, Payload State executes and payload is deployed.

OBC will always transition into Safe Mode if it detects a critical system failure.

5.2.5 Boot State

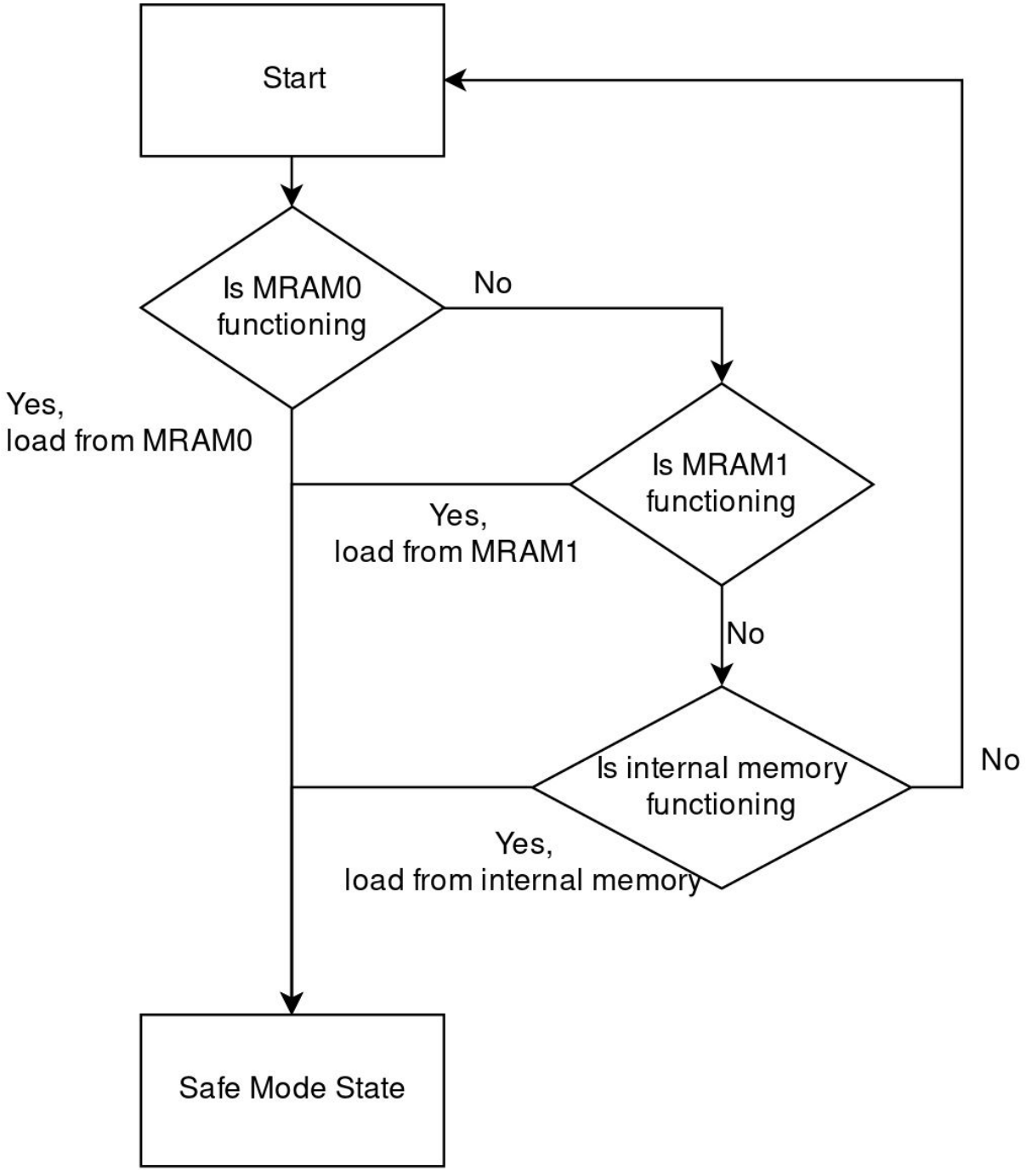

Figure 12. Boot State.

TMS570 will contain a small bootloader inside the NAND memory. The bootloader will be less than 5000 bits. Figure 12 shows the decision tree. If MRAM0 is functioning then it will load Safe Mode State software from MRAM0. Bootloader will also redirect accesses from internal NAND memory to MRAM0. MRAM1 and internal NAND memory also contain the same copies of software. TMS570 could also boot from MRAM1 and internal NAND memory.

5.2.6 Safe Mode State

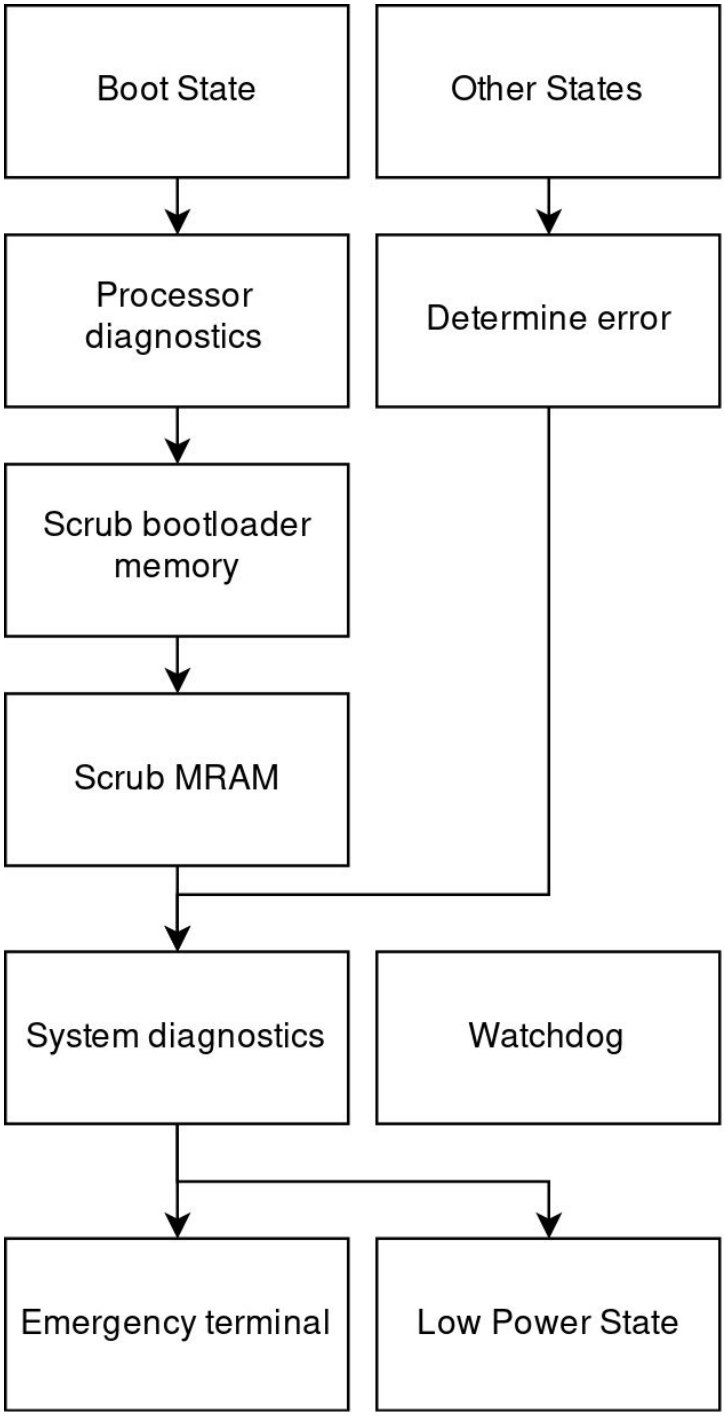

Figure 13. Safe Mode State.

Figure 13 show the decision tree of Safe Mode State. If Boot State enters Safe Mode State then it will perform microcontroller diagnostics and memory scrubbing. Microcontroller diagnostics checks the TMS570 microcontroller for hardware failures. Memory scrubbing protects the bootloader from data corruption.

Full system diagnostics involves checking each system for errors. Each subsystem in Homathko will have a self test to check capabilities.

If the error can not be resolved then it will open an emergency terminal. Emergency terminal will be enabled to for remote commands. The communications system will be active. The communications system will send out broadcasts containing the error message. Any ground station could contact Homathko in this state. Memory scrubbing and logging will also be active. All the other systems will be turned off.

5.2.7 Low Power State

In Low Power State, a few tasks are active. This is the default state. FreeRTOS will execute the tasks in bursts. Bursts conserve energy as it takes time to wake up from sleep. The tasks are list below.

Task	Description	Frequency (Hz)
Power control	Monitors the power flow of the Homathko. Determines if there is enough energy to keep Homathko functioning.	0.1
Inflight reprogramming	Manages the flashing of STM32 and TMS570. Allows new software to be reprogrammed while in flight.	0.5
Global Watchdog	Monitors all tasks to check if they are function correctly. Also monitors tasks on other systems.	0.5
CAN bus	Provides communication with CAN bus. Allows transmission and reception through CAN bus.	500000
Memory Scrubbing	Scrubs the memory to reduce bit errors.	0.1
Logging	Logs all data in the system.	1
Sun sensor	Monitors photodiodes.	1
Magnetometer	Monitors earth's magnetic field.	1
Thermopile	Monitors radiation from earth and sun.	1
Gyroscope	Monitors angular velocities.	1
Thermometer	Monitors temperatures.	1
ADCS	Attitude Determination Control System.	1

Table 12. Low Power State tasks descriptions for TMS570.

5.2.8 Communications State

The Communications State activate the communications system. Communications system allows the Homathko to transmit data to ground station. Communications State only

activates when Homathko is over the ground station. Communications State transitions to Low Power State when Homathko is not over ground station.

Task	Description	Frequency (Hz)
Power control	Monitors the power flow of the Homathko. Determines if there is enough energy to keep Homathko functioning.	0.1
Communications	Manages the communications stack. Contacts the ground station.	200000
Inflight reprogramming	Manages the flashing of STM32 and TMS570. Allows new software to be reprogrammed while in flight.	200000
Global Watchdog	Monitors all tasks to check if they are function correctly. Also monitors tasks on other systems.	0.5
CAN bus	Provides communication with CAN bus. Allows transmission and reception through CAN bus.	500000
Logging	Logs all data in the system.	1
Remote terminal	Executes remote commands from ground station.	200000
Sun sensor	Monitors photodiodes.	1
Magnetometer	Monitors earth's magnetic field.	1
Thermopile	Monitors radiation from earth and sun.	1
Gyroscope	Monitors angular velocities.	1
Thermometer	Monitors temperatures.	1
ADCS	Attitude Determination Control System.	1

Table 13. Communications State tasks descriptions for TMS570.

5.2.9 ADCS State

ADCS State powers on all sensors for data collection. This also includes GPS. ADCS will operate at 100 Hz to stabilize the satellite. ADCS State activates every few orbits to make slight adjustments.

Task	Description	Frequency

		(Hz)
Power control	Monitors the power flow of the Homathko. Determines if there is enough energy to keep Homathko functioning.	0.1
Global Watchdog	Monitors all tasks to check if they are function correctly. Also monitors tasks on other systems.	0.5
CAN bus	Provides communication with CAN bus. Allows transmission and reception through CAN bus.	500000
Logging	Logs all data in the system.	1
GPS	Determines exact position of satellite.	100
Accelerometer	Monitors acceleration.	100
Sun sensor	Monitors photodiodes.	10
Magnetometer	Monitors earth's magnetic field.	100
Thermopile	Monitors radiation from earth and sun.	10
Gyroscope	Monitors angular velocities.	100
Thermometer	Monitors temperatures.	1
ADCS	Attitude Determination Control System.	100

Table 14. Communications State tasks descriptions for TMS570.

5.2.10 Payload State

The Payload State is only used for executing the payload. TMS570 will monitor the execution of the payload using CAN bus. Payload State only activates when receiving a command in Communications State.

Task	**Description**	Frequency (Hz)
Power control	Monitors the power flow of the Homathko. Determines if there is enough energy to keep Homathko functioning.	0.1
Global Watchdog	Monitors all tasks to check if they are function correctly. Also monitors tasks on other systems.	0.5
CAN bus	Provides communication with CAN bus. Allows transmission and reception through CAN bus.	500000
Logging	Logs all data in the system.	1
Payload control	Executes the payload.	1000
Sun sensor	Monitors photodiodes.	10
Magnetometer	Monitors earth's magnetic field.	100
Thermopile	Monitors radiation from earth and sun.	10

Gyroscope	Monitors angular velocities.	100
Thermometer	Monitors temperatures.	1
ADCS	Attitude Determination Control System.	100

Table 15. Payload State tasks descriptions for TMS570.

5.2 Thermal Design

The components need to withstand temperatures from -20°C to 70°C. The table below shows the ratings of each component.

Component	Minimum Temperature(°C)	Maximum Temperature(°C)
TMS570LS3137PGE	-40	125
MMC3416xPJ	-40	85
ADT7320UCPZ-RL7	-40	125
MPU-6050	-40	85
SQW-110-01-F-D	-55	125
Novatel OEM719	-40	85
1P1G126QDBVRQ1	-40	125
TPS74701-Q1	-40	125
MR4A16BCMA35	-40	85
JTAG XDS100	-55	125
TMS320C5535AZHHA10	-40	85
T200F-010.0M	-40	85
SN74LVC1G17DSFR	-40	125
LTC6088HDHC#PBF	-40	125
AD7903BRQZ	-40	125
CM13032768DZFT	-40	85
MCP4921-E/MC	-40	125
THS4532IPW	-40	125
TPS7A7200RGTR	-40	125
TLV7113333DDSER	-40	125
LP3996SD-2533/NOPB	-40	85
TPS62170DSGR	-40	85
TPS7A8701RTJR	-40	125

Table 16. Maximum and minimum temperatures of components.

Table 16 shows each component's temperature ratings. All of the components satisfy the temperature requirement in space environment. If the temperature exceeds the ratings then temperature sensor would warn the TMS570. TMS570 could turn off parts to reduce temperature.

5.3 RF Shielding Design

OBC will have a RF shield surrounding the whole OBC PCB. The RF shielding will use cold reel steel with tin plating.

5.4 Power Design

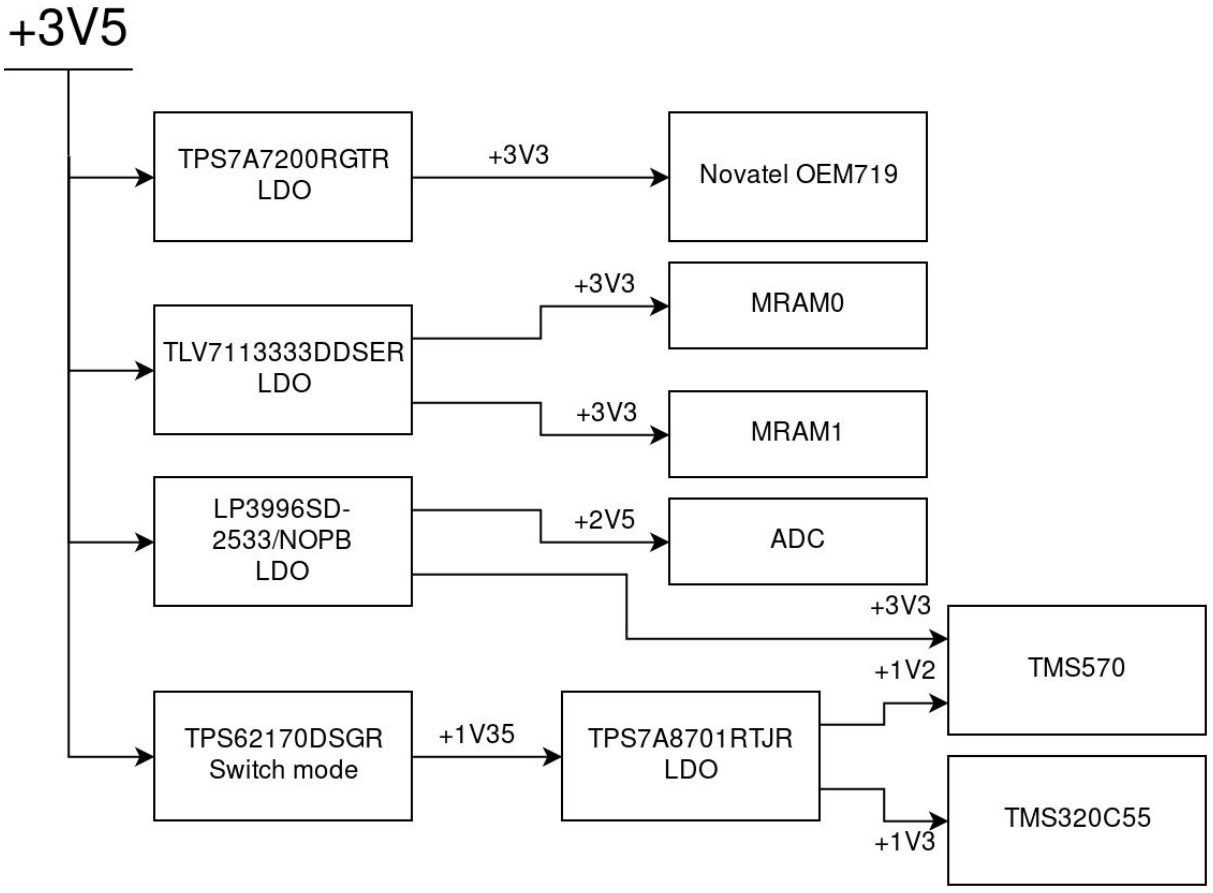

Figure 15. Overview of power components for OBC.

The linear LDO TPS7A7200RGTR [17] drops 3.5V to 3.3V for Novatel OEM719. The linear LDO TLV7113333DDSER [18] drops 3.5V to 3.3V for MRAM. However, TLV7113333DDSER has power switches to turn MRAM off and on. Linear LDO LP3996SD-2533/NOPB [19] drops 3.5V to 3.3V and 3.5V to 2.5V. TPS62170DSGR [20] is a switch mode regulator with an efficiency of 90%. It drops 3.5V to 1.35V and passes 1.35V into a low pass. Linear LDO TPS7A8701RTJR drops 1.35V to 1.2V and 1.35V to 1.3V. Power components were seperated to reduce power supply noise. For example, PSRR for TPS7A7200RGTR is 40 db. Noise from Novatel OEM719 will not affect other components.

Voltage Regulation Components	Voltage Input (V)	Voltage Output (V)	Typical Current (mA)	Maximum Current (mA)	Efficiency	Typical Power (mW)	Maximum Power (mW)
TPS62170DSGR	3.5	1.35	260	500	0.9	39	75

Table 17. Switch mode component power dissipation.

Voltage Regulation Components	Voltage Input (V)	Voltage Output (V)	Typical Current (mA)	Maximum Current (mA)	Typical Power (mW)	Maximum Power (mW)
TPS7A7200RGTR	3.5	3.3	400	1700	80	340
TLV7113333DDSER	3.5	3.3	120	240	24	48
LP3996SD-2533/NOPB	3.5	3.3	100	200	20	40
TPS7A8701RTJR	1.35	1.2	220	440	33	66
TPS7A8701RTJR	1.35	1.3	30	40	1.5	2
Total					158.5	496

Table 18. Linear LDO components power dissipation.

Component	Minimum Voltage (V)	Typical Voltage (V)	Maximum Voltage (V)
TMS570LS3137PGE VCC,VCCPLL	1.14	1.2	1.32
TMS570LS3137PGE VCCIO	3	3.3	3.6
TMS570LS3137PGE VCCAD	3	3.3	3.6
TMS570LS3137PGE VCCP	3	3.3	3.6
TMS570LS3137PGE VADREFHI	0	3.3	3.6
TMS570LS3137PGE VADREFLO	0	3.3	3.6
Novatel OEM 719 VCC	3.135	3.3	3.465
Novatel OEM 719 LNA	0	0	0
MPU-6050 VDD	2.375	3.3	3.46
MPU-6050 VLOGIC	1.7	3.3	3.46
MMC3416xPJ VDA	1.62	3.3	3.6
MMC3416xPJ VDD	1.62	3.3	3.6
ADT7320	2.7	3.3	5.5
TPL5010-Q1	1.8	3.3	5.5
1P1G126QDBVRQ1	1.65	3.3	5.5
TMS320C5535AZHHA10 CVDD	1.24	1.3	1.43

TMS320C5535AZHHA10 DVDDIO	2.97	3.3	3.63
T200F-010.0M	3.135	3.3	3.465
SN74LVC1G17DSFR	1.5	3.3	5.5
LTC6088HDHC#PBF	2.7	3.3	5.5
AD7903BRQZ VDD		2.5	
AD7903BRQZ VIO		3.3	
MCP4921-E/MC		3.3	
THS4532IPW		3.3	
MR4A16BCMA35	3.3	3.3	3.3

Table 19. Voltages of components.

Component	Minimum current (mA)	Typical Current (mA)	Maximum Current (mA)
TMS570LS3137PGE VCC,VCCPLL	0.1	220	440
TMS570LS3137PGE VCCIO	0.1		10
TMS570LS3137PGE VCCAD	0.1	0	0
TMS570LS3137PGE VCCP	0.1		60
TMS570LS3137PGE VADREFHI	0.1	0	
TMS570LS3137PGE VADREFLO	0.1	0	
Novatel OEM 719 VCC	0.1	364	
Novatel OEM 719 LNA	0	0	0
MPU-6050 VDD	0.1	3.8	4
MPU-6050 VLOGIC	0.1	0.1	
MMC3416xPJ VDA	0.001	0.14	
MMC3416xPJ VDD	1	1	
ADT7320		0.21	0.265
TPL5010-Q1	0.1	0.2	0.4
1P1G126QDBVRQ1		5	24
TMS320C5535AZHHA10 CVDD		30	
TMS320C5535AZHHA10 DVDDIO		1	
T200F-010.0M		2.1	2.1
SN74LVC1G17DSFR		5	
LTC6088HDHC#PBF		5	
AD7903BRQZ VDD		5.8	
AD7903BRQZ VIO		1	

MCP4921-E/MC		2	
THS4532IPW		5	
MR4A16BCMA35	10	106	

Table 20. Currents of components.

Component	Typical Power (mW)	Maximum Power (mW)	Typical duty cycle	Apparent Typical Power (mW)
TMS570LS3137PGE VCC,VCCPLL	264	580.8	0.5	132.057
TMS570LS3137PGE VCCIO		36	0.5	0.15
TMS570LS3137PGE VCCAD		0	0.5	0.15
TMS570LS3137PGE VCCP		216	0.5	0.15
TMS570LS3137PGE VADREFHI				0
TMS570LS3137PGE VADREFLO				0
Novatel OEM 719 VCC	1201.2		0.05	60.357825
Novatel OEM 719 LNA	0	0	0.05	0
MPU-6050 VDD	12.54	13.84	0.1	1.46775
MPU-6050 VLOGIC	0.33	0	0.1	0.186
MMC3416xPJ VDA	0.462	0	0.1	0.047658
MMC3416xPJ VDD	3.3	0	0.1	1.788
ADT7320	0.693	1.4575		0
TPL5010-Q1	0.66	2.2		0.18
1P1G126QDBVRQ1	16.5	132	0.05	0.825
TMS320C5535AZHHA10 CVDD	39		1	39
TMS320C5535AZHHA10 DVDDIO	3.3		1	3.3
T200F-010.0M	6.93		1	6.93
SN74LVC1G17DSFR	16.5		1	16.5
LTC6088HDHC#PBF	16.5		1	16.5
AD7903BRQZ VDD	12		1	12
AD7903BRQZ VIO	3.3		1	3.3
MCP4921-E/MC	6.6		1	6.6

THS4532IPW	16.5		1	16.5
MR4A16BCMA35	349.8		0.5	191.4
Total				509.389233

Table 21. Power dissipation of components.

$$P_{average} = P_{components} + P_{LinearLDOs} + P_{SwitchMode}$$
$$P_{average} = 0.5W + 0.159W + 0.039W = 0.698W$$

As shown in Table 17, Table 18, and Table 21, on average OBC consumes 0.6W to 0.7W of power. Power may increase when GPS is turned on. GPS consumes 1.2W of power when turned on. MRAM and TMS570 consume most of the power on OBC when taking duty cycle into account.

5.5 Computation Fault Tolerance Design

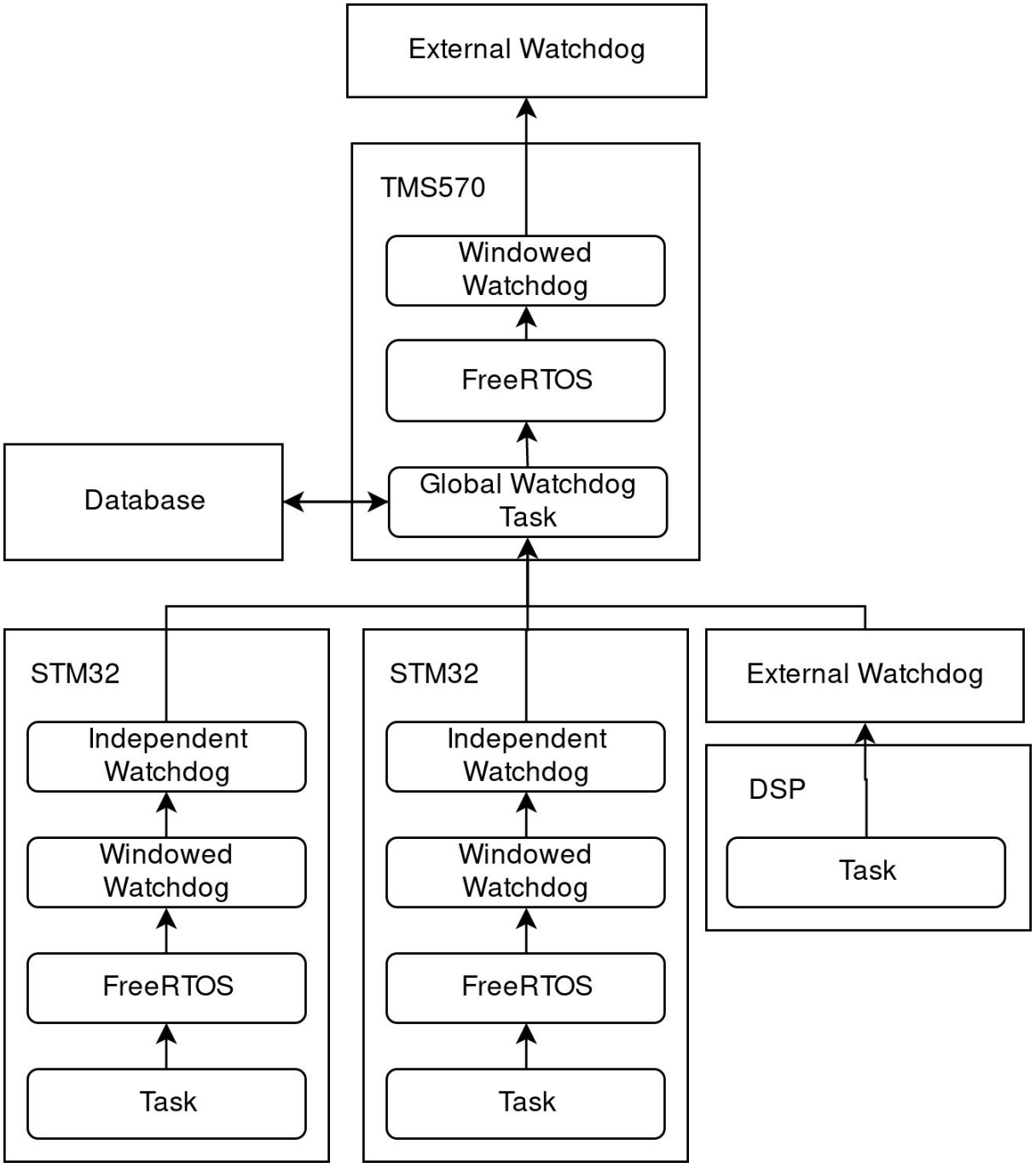

Figure 15. Watchdogs in Homathko.

TMS570 has a dedicated external watchdog as shown in Figure 15. External watchdog will check if TMS570 is functioning. TMS570 also has an internal watchdog. Internal watchdog will monitor FreeRTOS. FreeRTOS will monitor each task for failure. When a task fails, FreeRTOS restarts the task automatically. Global watchdog task runs on the TMS570. Global watchdog task monitors all other MCUs and DSPs for check if they respond. Global watchdog task has multiple countdown timers. There is a countdown timer for each of the MCUs and DSPs. The countdown timer resets every time the MCUs or DSPs communicates

with TMS570. If the countdown timer reaches zero then TMS570 will attempt to communicate to the MCUs or DSPs. If communication fails then TMS570 will force a hard reset and reflash the firmware on the MCUs or DSPs. This increases the reliability of the MCUs and DSPs.

STM32 has two internal watchdogs for redundancy. These watchdogs will reset the STM32 if it fails to respond.

The probability of a watchdog succeeding is greater than 0.85 [21]. Calculations assume average conditions in space. Assume reliability of an internal watchdog is equal to an external watchdog.

The external watchdog at the top.

$$P_{EWDG} = 0.85$$

The windowed watchdog inside the TMS570.

$$P_{WWDG} = 0.85$$

The hardware reliability of TM570.

$$P_{TMS570} = P_{EWDG} + P_{WWDG} - P_{EWDG} \cdot P_{WWDG}$$

$$P_{TMS570} = 0.9775$$

The reliability of FreeRTOS.

$$P_{FreeRTOS} = 0.7$$

The reliability of a task in FreeRTOS.

$$P_{Task} = P_{FreeRTOS} + P_{TMS570} - P_{FreeRTOS} \cdot P_{TMS570}$$

$$P_{Task} = 0.9933$$

Single tasks has a probability of at least 0.9933 for reliability. This is sufficient for all tasks in TMS570. Reliability is defined as the probability of any watchdog recovering the task from a fault.

5.6 Sensor Fault Tolerance Design

Sensor data should be verified. Firstly, sensor data should be in the range of possible sensor values. Secondly, sensor data's timestamps should be a valid time. Timestamps should also be retrieved in chronologic order. Lastly, If multiple sensor messages are corrupted then the sensor producing them will be ignored. This is to protect against sensor failure.

5.7 PCB Design

The PCB design will use 6 layers for routing components. There are 4 BGA components on the PCB. The 6 layers are required to route 300 pads on the TMS570'S BGA.

Layer	Type
1	Signal
2	Power

3	Signal
4	Signal
5	Ground
6	Signal

Table 22. Layout PCB.

The PCB will have the layout as shown in Figure 22. There are 4 signal layers used to fan out the TMS570's BGA. Each signal layer fans out two layers of pads. This would get the traces out of the BGA. TMS570 requires +3.3V and +1.2V. Therefore, a power layer is needed. Power is delivered through a layer only reserved for power. Ground layer also provides ground to the TMS570. Power and ground shield signals from noise. There are 2 signal layers in between power and ground. Signals going from MRAM to TMS570 uses those 2 signal layers. This would protect the signals going from noise.

6.0 Testing and Verification

6.1 Hardware Testing and Verification

The hardware testing includes testing from CSDC. CSDC will conduct thermal-vacuum, vibration, radiation, and magnetic field testing. Tests outside of CSDC include:

- Power consumption
- ADC/DAC SNR
- CAN bus
- Watchdog timer
- Sensor drift

Power consumption testing will measure the power consumption at full load. This is test if the power system is capable of handling full load. All power supply voltages and currents will be monitored. ADC/DAC SNR will be measured to ensure the communications system will have sufficient link margin.

CAN bus messages will be tested to ensure the CAN bus has full throughput. Moreover, CAN bus messages will be measured by a CAN bus probe. Watchdog timers will be artificially activated to ensure they reset the microcontroller properly. Sensors will be measured to ensure they won't drift over time and temperature.

6.2 Software Testing and Verification

The software testing and verification will consist of automated and manual testing and verification. The following software tools will be used:

- Cppcheck
- Unity testing framework
- CMock
- Gitlab continuous integration
- Tracealyzer

- OpenOCD

6.2.1 Automated Testing and Verification

Automated testing will be the first line of defense against bugs.

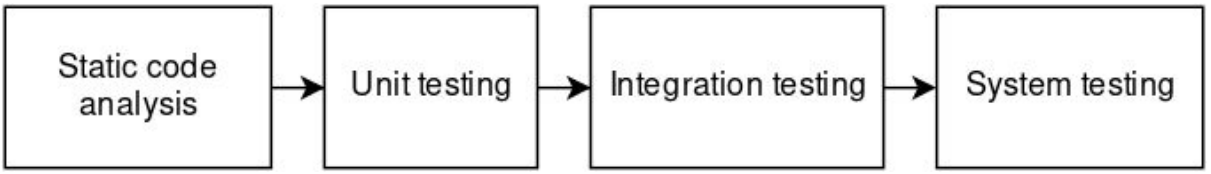

Figure 16. Flowchart of automated testing.

Static code analysis examines the source code of the software. It checks the syntax of the source code to find bugs. Cppcheck will be used as a static code analysis for C. Cppcheck was chosen because it was lightweight and easy to use.

Unit testing, integration testing, and system testing will use the Unity testing framework. Unity was chosen because it supports embedded systems and is lightweight. Unit testing will test each individual function in the source code. Integration testing will test the interactions between each function. System testing will test the entire software system with other systems. Automated tests will contain premade criteria for the software to pass.

System testing will also include:

- Performance testing
- Fault tolerance testing
- Simulation of entire satellite

CMock can be used to mock functions. If Function A is incomplete and is a dependency for Function B then CMock can be used. CMock mocks Function A and allows Function B to be tested. CMock would be useful in integration testing and system testing.

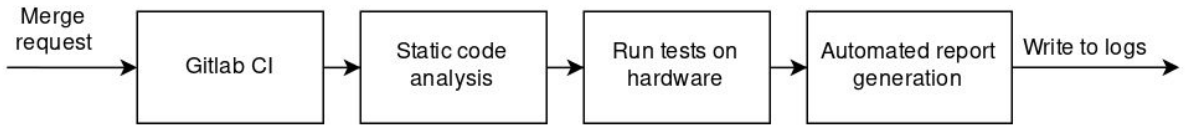

Figure 17. Gitlab continuous integration automated flow.

Figure 2 shows gitlab continuous integration. When a merge request arrives, gitlab continuous integration starts static code analysis. Then the software is tested on real hardware. Finally, a report is generated based on the tests performed. If a single test has failed then the merge request is rejected.

6.2.2 Manual Testing and Verification

After performing automatic testing, manual testing will begin.

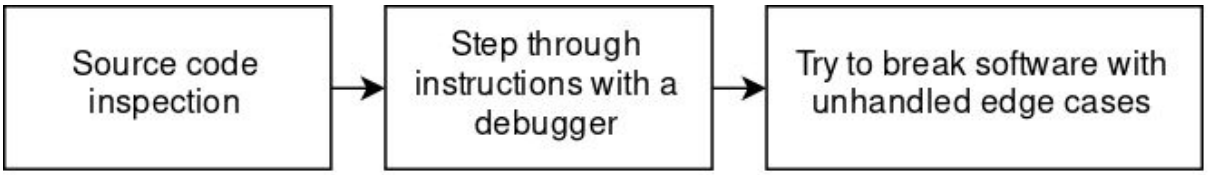

Figure 18. Manual testing.

Firstly, source code is inspected manually to ensure it is functioning. Secondly, a debugger is used. OpenOCD and Tracealyzer were chosen to debug software. A debugger allows the user to check the software's memory at a specific time. Therefore, software's memory can be examined to ensure the software is reading and writing correct data. Lastly, the software will be given edge cases to test error handling. If the software did not have those specific cases then the software's behaviour will be unpredictable.

7.0 References

[1] "Thermal-Vacuum Test pg. 15", *http://csdcms.ca*, 2015. [Online]. Available: http://csdcms.ca/wp-content/uploads/2016/10/CSDC_DIETR_3a_RELEASED_2014-10.pdf. [Accessed: 15- Apr- 2017].
[2] "Outgassing Data for Selecting Spacecraft Materials System", http://nasa.gov, 2017. [Online]. Available: http://outgassing.nasa.gov. [Accessed: 15- Apr- 2017].
[3] "Launch Environment Tests pg. 14", *http://csdcms.ca*, 2015. [Online]. Available: http://csdcms.ca/wp-content/uploads/2016/10/CSDC_DIETR_3a_RELEASED_2014-10.pdf. [Accessed: 15- Apr- 2017].
[4] C. Hussmann, "Error Control Code Evaluation and Design for the ECOSat-II Cube Satellite", MASc, University of Victoria, 2016.
[5] "MMC3416xPJ pg. 1", *http://www.memsic.com*, 2013. [Online]. Available: http://www.memsic.com/userfiles/files/Datasheets/Magnetic-Sensors-Datasheets/MMC3416xPJ_Rev_C_2013_10_30.pdf. [Accessed: 15- Apr- 2017].
[6] "MPU-6000-Datasheet pg. 12", *https://www.invensense.com*, 2015. [Online]. Available: https://www.invensense.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf. [Accessed: 15- Apr- 2017].
[7] "OEM719 Datasheet pg. 2", *https://www.novatel.com*, 2015. [Online]. Available: https://www.novatel.com/assets/Documents/Papers/OEM719-Product-Sheet.pdf. [Accessed: 15- Apr- 2017].
[8] "TPL5010-Q1 Datasheet pg. 11", *https://www.ti.com*, 2015. [Online]. Available: http://www.ti.com/lit/ds/symlink/tpl5010-q1.pdf. [Accessed: 15- Apr- 2017].
[9] "MR4A16B Datasheet ", *https://www.everspin.com*, 2017. [Online]. Available: https://www.everspin.com/file/162/download. [Accessed: 15- Apr- 2017].
[10] Jason Heidecker
"MRAM Technology and Status"
Jet propulsion Lab, California Institute of Technology, 2012

[11]X. Zhang, Q. Guo, Y. Li, C. He and L. Wen, "Total Ionizing Dose and Synergistic Effect of Magnetoresistive Random Access Memory", *Chinese Physics C*, 2016.

[12] "tx350 Datasheet ", *https://www.*conwin.*com*, 2017. [Online]. Available: http://www.conwin.com/datasheets/tx/tx350.pdf. [Accessed: 15- Apr- 2017].
[13] "AD7903 Datasheet ", *http://www.analog.com*, 2015. [Online]. Available: http://www.analog.com/media/en/technical-documentation/data-sheets/AD7903.pdf. [Accessed: 15- Apr- 2017].
[14] "MCP4921 Datasheet ", *https://www.microchip.com*, 2010. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/22248a.pdf. [Accessed: 15- Apr- 2017].
[15] "TMS320C5535 Datasheet ", *https://www.ti.com*, 2015. [Online]. Available: http://www.ti.com/lit/er/sprz373c/sprz373c.pdf. [Accessed: 15- Apr- 2017].
[16] "TMS570LS3137 Datasheet ", *https://www.ti.com*, 2015. [Online]. Available: http://www.ti.com/lit/gpn/tms570ls3137. [Accessed: 15- Apr- 2017].
[17] "TPS7A7200 Datasheet ", *https://www.ti.com*, 2015. [Online]. Available: www.ti.com/lit/ds/symlink/tps7a7200.pdf. [Accessed: 15- Apr- 2017].
[18] "TLV7113333DDSER Datasheet ", *https://www.ti.com*, 2010. [Online]. Available: http://www.ti.com/lit/ds/symlink/tlv711.pdf. [Accessed: 15- Apr- 2017].
[19] "LP3996 Datasheet ", *https://www.ti.com*, 2013. [Online]. Available: http://www.ti.com/lit/ds/symlink/lp3996.pdf. [Accessed: 15- Apr- 2017].
[20] "TPS6217x Datasheet ", *https://www.ti.com*, 2017. [Online]. Available: http://www.ti.com/lit/ds/symlink/tps62170.pdf. [Accessed: 15- Apr- 2017].
[21] Jacob Beningo and Ann Arbor, "A Review of Watchdog Architectures and their Application to Cubesats", *AERO590*, 2010.