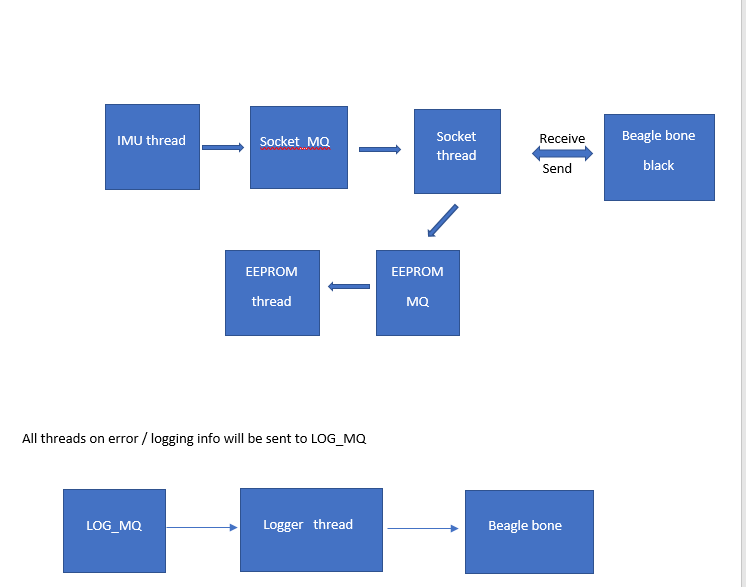
# Flight Simulator

TIVA1294



All threads on error / logging info will be sent to LOG\_MQ

Logger thread

Beagle bone

LOG\_MQ

Monitor thread

Monitor thread

(Updates an array)

Task Notify from all threads

Sequencer thread

(monitors the array) for system health

Blink LED

TIVA 1294

Main.c – will create all threads and corresponding threads will initialize and wait on respective message queues.

IMU thread – Initialize and wait on IMU\_MQ and read data from sensor and send to SOCKET\_MQ. Any error will be pushed to LOG\_MQ

Socket thread – Initialize and wait on SOCKET\_MQ. When data is received to transmit send it over UART

EEPROM thread – wait on EEPROM MQ and If the message received from BBG is fatal then add data FATAL message to EEPROM.

Logger thread - wait on LOG\_MQ When data is received to transmit send it over UART.

Monitor thread – monitor thread receives task notify from all threads and thread updates the system status array which will be monitored by the sequencer thread

Sequencer thread – The thread periodically checks the system health array updated by the monitor thread.

Beagle Bone Black

Led thread

LED\_MQ

**LOG\_MQ**

**Logger thread**

SEND\_MQ

Processor thread

Socket thread

(Server)

Instance 2

PROCESS\_MQ

**TIVA**

Monitor thread

LED Driver

(Blink LED)

Heartbeat thread(Updates an array)

Conditional signal from all threads

Beagle Bone Green

Socket thread: Instance 1– accept connection for Tiva and push to PROCESS\_MQ. Then it blocks on SEND\_MQ and pulls processed data and acknowledges the client of the state. If state is fatal, the tiva logs into EEPROM.

Socket thread: Instance 2 – accept connection from tiva to log Tiva status and push it to LOG\_MQ.

Processor thread – wait on PROCESS\_MQ and pull raw IMU data. Perform sensor fusion, update state as normal and fatal and push to push to SEND\_MQ and LED\_MQ.

Logger thread: Instance 2- blocks on LOG\_MQ and pulls tiva log data and writes it to logger file.

Led thread: Blocks on LED\_MQ and pulls LED states that should be outputted to the LEDs through LED driver.