CySat: Satellite Mission Design

DESIGN DOCUMENT

Team Number: sdmay21-25
Client: M2I
Adviser: Dr. Phillip Jones
Team Members/Roles:
Alexis Aurandt - OBC Lead and Payload Sub-Lead
Alex Constant - Ground Station Front-End Lead and Voltage Boost Board Sub-Lead
Chandler Jurenic - Payload Lead and OBC Sub-Lead
Jeffrey Richardson - ADCS Lead/Team Lead
John Lenz - Radio Lead
Scott Dressler - EPS Lead and Voltage Boost Board Sub-Lead
Team Email: cysat-f2020@iastate.edu
Team Website: http://sdmay21-25.sd.ece.iastate.edu

Revised: October 25th/2.0

Executive Summary

Development Standards & Practices Used

- NASA and CubeSat hardware standards
- Consistent code commenting and documentation for software.
- Code reviews by team before merges
- UART
- I2C
- Python 3+ for Ground Station

Summary of Requirements

- Needs to power up after been deployed from the International Space Station
- Needs to stabilize and point itself towards earth
- Needs to take soil moisture readings from Earth via a microwave radiometer
- Needs to transmit data back to the ground station in Ames, IA
- Needs to be able to collect data for its orbit life (6 months)
- Needs to meet NASA's CubeSat requirements

Applicable Courses from Iowa State University Curriculum

CPR E 288: Embedded Systems

CPR E 488: Embedded Systems Design

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

Table of Contents

List of Figures/Tables/Symbols/Definitions

ADCS – Altitude Determination and Control System

EPS – Electrical Power System

M:2:I – Make to Innovate

OBC – On-board Computer

SDR – Software Defined Radio

UHF Radio- Ultra High Frequency Radio

CubeSat – A standard, miniaturized scientific satellite comprising connected 10 x 10 x 10 cm cubes

1 Introduction

1.1 Acknowledgement

We would like to thank last year's senior design team for providing handoff documentation, providing guidance, and for being available for questions. Additionally, we would like to thank Dae-Young Lee for his expertise and his guidance with the ADCS sub-system. Finally, we want to thank Dr. Jones for meeting with us weekly and for giving us guidance.

1.2 Problem and Project Statement

The CySat is a cube satellite, which is a standardized form of miniaturized scientific satellite. CySat will be deployed from the International Space Station, after which it will orbit the earth for approximately 6 months with the goal of collecting and relaying soil moisture data back to our ground station in Howe Hall on the Iowa State University campus.

The CySat is a student project started and operated by M:2:I. The sole purpose of this project is to get students engaged in a hands-on project, and the driving force behind the project is students wanting to develop a satellite that will be launched into space. Originally, this project was only for Aerospace engineering students, but M:2:I soon realized they needed Computer, Electrical, and Software engineers to take care of the onboard electronics. This is where our senior design team comes in. There are many different subsystems that need our expertise.

The CySat comprises six subsystems. These subsystems control the satellite's orientation with respect to the Earth, collect and process data, and stream the data back to Earth during communications windows. The subsystems are as follows:

- The OBC communicates with all the other subsystems and ensures that the satellite is operating according to specifications.
- The ADCS stabilizes and points the satellite toward Earth.
- The EPS regulates power from the solar cells.
- The radio relays data and commands between the CySat and Earth.

- The ground station receives data, sends commands, and acts as the interface between the M:2:I satellite team and the satellite.
- The payload uses an SDR to gather soil moisture readings from earth.

1.3 Operational Environment

The CySat will be launched into space, and it will orbit the Earth. This necessitates that the internal hardware of the CySat, as well as software running on that hardware, are robust and capable of failure recovery with minimal loss.

1.4 Requirements

1.4.1 Functional Requirements

- Must power up after been deployed from the International Space Station
- Must stabilize and point itself towards earth
- Must take soil moisture readings from Earth via a microwave radiometer
- Must transmit data back to the ground station in Ames, IA
- Must collect data for its orbit life (6 months)
- Must meet NASA's CubeSat standards and regulations
- Must receive and execute commands issued by the Ground Station
- Must successfully deorbit at the end of its lifespan

1.4.2 Nonfunctional Requirements

- Ground Station UI is performant

1.5 Intended Users and Uses

Our senior design team and the M:2:I team are the end users for this CySat. This CySat is a creative project for students to be able to launch a satellite into space and receive data back.

1.6 Assumptions and Limitations

1.6.1 Assumptions

- We assume correct installation of the sub system by M2I.
- Software on the CySat is of the same version as what we used to implement the subsystem functionality
- Databases for the Ground Station are not yet implemented

1.6.2 Limitations

- The hardware and software of the CubeSat must comply with NASA regulations as well as CubeSat standards, and the hardware must fit within a 10 x 10 x 10 cm cube in the satellite housing.
- Few operation times in lab due to COVID regulations
- Mock launches are the only feasible way of testing the subsystems

1.7 Expected End Product and Deliverables

The hardware and software for a CubeSat will be delivered to M:2:I, who will hand the CubeSat off to CubeSat testing for prelaunch testing, after which the CubeSat will be launched to the International Space Station for deployment. The deliverables for this project are each of the subsystems; the expectations for each, as well as their tentative due dates, are laid out below.

The Ground Station (April 2020)

The Ground Station software will be installed on a desktop device in Howe Hall. The Ground Station will allow the user to visualize the satellite's location, send commands to the satellite, and receive and store data from the satellite.

The Payload SDR (April 2020)

The payload's SDR system will be connected to the satellite via the OBC using UART. This SDR will be capable of collecting data with the radiometer and capable of transmitting the data back to the OBC. Though most of the subsystem will be partially completed throughout the Fall 2020 and Spring 2021 semesters, the finished payload subsystem will be ready by April 2020.

The ADCS (April 2020)

The ADCS must be able to switch between modes of operations such as active and passive detumbling. ADCS will send telemetry data to the OBC to be recorded / sent to the ground station via radio. ADCS will have designated modes of operation for gathering scientific information and transmitting to the ground station based off of orbit.

The UHF Radio (April 2020).

UHF Radio subsystem must be able to receive instructions from the ground station. Take the instructions and communicate instructions to OBC. OBC will then send the requested data back to the ground station through the UHF Radio.

The EPS (March 2020)

The EPS provides energy to the rest of the satellite and reports its heath to the OBC. It controls these systems differently based on the satellite's current mode of operation.

The OBC (April 2020)

The OBC is the heart of the entire CySat. The OBC must be able to communicate to all the different subsystems. It must be able to efficiently read messages and give commands in order to achieve the CySat's purpose of successfully relaying moisture data back to Earth.

Handoff Documentation (May 3rd)

If this project does not get done on time for whatever reason, the responsibility to finish the project will be on next years' CySat team. If we get into the Spring 2021 semester and realize we don't have enough time, handoff documentation must be created.

2 Project Plan

2.1 Task Decomposition

We will be decomposing tasks by subsystem. Each numbered task is a known requirement of the subsystem, and the lettered subtasks below are tentative subtasks for implementing the required functionalities.

2.1.1 Ground Station

1. Ground Station Communication
    a. Ground Station receives packets
    b. Ground Station interprets packets
    c. Ground Station sends packet
2. Ground Station Data and Command Logging
    a. Create a logging schema for Command logging
    b. Ground station logs commands sent and received on a day to day basis
3. Ground Station Custom Commands
    a. Create custom command schema for Ground Station
    b. Implement custom commands
4. Ground Station Visualization
    a. Subsystem Health Check Visualization
    b. Ground Station visualizes satellite's location around the Earth

2.1.2 UHF Radio

1. Radio Communication
    a. Receive and send packets from computer to radio for debugging
    b. Receive and send health check to OBC
    c. Receive and send commands to OBC
    d. Receive and send beacon and packets to Ground Station
    e. Receive and send packets from Ground Station and OBC

2.1.3 EPS

1. EPS Communication
    a. Receive and send commands to OBC
    b. Send health checks to OBC
2. EPS heath checks
    a. Monitor battery temperature
    b. Check voltage and current inputs and outputs
    c. Verify charging status
    d. Track energy levels
3. EPS charge and discharge
    a. EPS can be charged by M:2:I solar cells
    b. EPS can power CySat subsystems
4. EPS battery protection
    a. Battery temperature control
    b. Short-circuit/over-discharge/over-charge protection

2.1.4 Payload

1. Payload Communication
    a. Send and Receive data to/from OBC
    b. Send Radiometer data to OBC
    c. GNU radio to SDR
    d. UART functionality
    e. I2C to LNA board
2. Payload Data Collection
    a. Packet Protocol functionality
    b. OBC transfers data through UHF antenna
    c. Capture mode needs to be completed and tested
    d. Radiometer program to run at startup of Linux subsystem
3. Payload Functionality
    a. Must be powered by EPS
    b. SDR will require OBC to use data collected
    c. Update time
    d. Operated by OBC

2.1.5 ADCS

1. Storing Telemetry data when out of ground station range
2. Mode activation control
    a. Passive Detumble
    b. Active Detumble
    c. Active Comms Control
3. Set magnetometer configuration (compute magnetometer offset and sensitivity matrix)
4. Y-Wheel Ramp-up Test
5. Y-momentum mode commissioning 1 & 2
6. Sun/Nadir sensor test
7. Active comms control
8. 8-bit Health check
    a. Communication with OBC
    b. X,Y,Z rates
    c. Current Altitude
    d. Current Sun sensor
    e. Nadir sensor
    f. Magnetometer deployment
9. Re-entry

2.1.6 OBC

1. OBC Communication
    a. Receive and send info to computer for debugging
    b. Receive and send packets to Ground Station
    c. Receive and send commands to EPS
    d. Receive and send commands to UHF Radio
    e. Receive and send commands to ADCS
    f. Receive and send commands to SDR Payload
    g. Store payload information on file system
2. OBC Optimization
    a. Interrupt based I2C and UART
    b. FreeRTOS investigation

3. OBC Live Updates
    a. Create bootloader that allows live patches
    b. Receive live patches from Ground Station
4. OBC Land Tests
    a. Implement Mock Mock Launch
    b. Implement Mock Launch
    c. Implement Mock Mission

2.1 Risks And Risk Management/Migration

Many of the risks associated with this project are difficult to mitigate, as they are related to the satellite's performance in space. The mitigation for these types of risks is to test extensively, consistently, and well. Below are a few of the risks our team has chosen to identify based on current knowledge of the CySat project.

Risk	Explanation	Estimated Probability
Loss of communication with Ground Control	Risk of losing communication with the ground control station based off of the tumbling of the CySat	10% Mitigated by having passive detumbling modes for when communication cannot be established
Difficulties setting two-way communication with a single radio module	Radio can be set as a transceiver on the CySat, but previous teams have had difficulty with this configuration	60% Mitigated by installing a second radio system on the CySat.
Connecting OBC to other subsystems	When combining separate subsystems there is a possibility of the connection messing up already working functionalities	50% Since the 6 subsystems are communicated through Git, we can just rollback the error the connection caused.
Loss of power	Battery operated CySat could lose power mid operation.	20% CySat has on board solar panels to recharge its battery EPS sends battery status to OBC
Task exceeds expected time	We expect sometime in the project to have a task or tasks which will need to be re-evaluated based off of the difficulty they are currently presenting.	90% Mitigated by using the scrum workflow and by communication blockers throughout the team / Dr. Jones to solve pending issues.

2.2 Project Proposed Milestones, Metrics, and Evaluation Criteria

The milestones and metrics associated with the CySat satellite are based on preliminary estimations of performance. Where numbers are unknown, an 'x' is used to delineate the unknown number – these values are to be determined through discussion with M:2:I and iteration on the project deliverables.

Milestones are roughly grouped by their associated subsystem, and follow below:

Milestone: Ground Station Receives OBC Communication

Metrics: Packets sent/received per second

Evaluation Criteria: Ground Station will successfully handle x packets per second with no loss due to concurrency

Milestone: Ground Station Sends OBC Communications

Metrics: Packets sent/second, confirmation of receipt of packets

Evaluation Criteria: Ground station may send up to x packets/second, confirms receipt of packet for x% of packets

Milestone: UHF Radio Communicates with OBC

Metrics: Messages will be transmitted to the OBC send back to the UHF Radio, and printed correctly through putty.

Evaluation Criteria: UHF will successfully transmit and receive data from the OBC with a packet loss percentage below some percentage

Milestone: UHF Radio Communicates with Ground Station

Metrics: Messages will be transmitted to the Ground Station send through the UHF Radio, and back to the Ground Station, being printed correctly back through the ground station UI.

Evaluation Criteria: Messages will be transmitted to the Ground Station send through the UHF Radio, and back to the Ground Station

Milestone: EPS Health Check

Metrics: When prompted, the EPS sends its own health data to the OBC. This data will contain the temperature of the batteries in Celsius, the input and output voltages in Volts, the input and output currents in mili-Amperes, and the battery capacities.

Evaluation Criteria: All included measurements are accurate (including charging and discharging) and can be received by the OBC.

Milestone: Payload's SDR Communicates with OBC

Metrics: The OBC will use UART to utilize the SDR to receive data that the SDR collected using the radiometer application.

Evaluation Criteria: The OBC successfully can display the information that the SDR got from the radiometer.

Milestone: ADCS Communicates with OBC

Metrics: The OBC will use I2C to send commands to the ADCS for transitioning between operational modes. The ADCS will send telemetry data to the OBC to be recorded.

Evaluation Criteria: The OBC successfully can display and store telemetry data from ADCS. OBC will be able to switch modes of operation on the ADCS through I2C.

Milestone: ADCS has multiple operational modes defined

Metrics: Code for operational modes are implemented, such as active and passive detumble.

Evaluation Criteria: Operational modes are defined based off of the operation mode control flow for ADCS. Each operation modes takes in telemetry data as well as OBC command data.

Milestone: OBC Optimization

Metrics: Multithreading through FreeRTOS and/or creating interrupts for UART and I2C will increase the speed and response time of the OBC to outside communication from other subsystems.

Evaluation Criteria: The response time of the OBC will increase by ~50%

Milestone: OBC Live Updates

Metrics: The Ground Station will give live patches to OBC through the UHF radio. The Ground Station will use a bootloader to successfully update.

Evaluation Criteria: The OBC will successfully accept a live patch from the Ground Station.

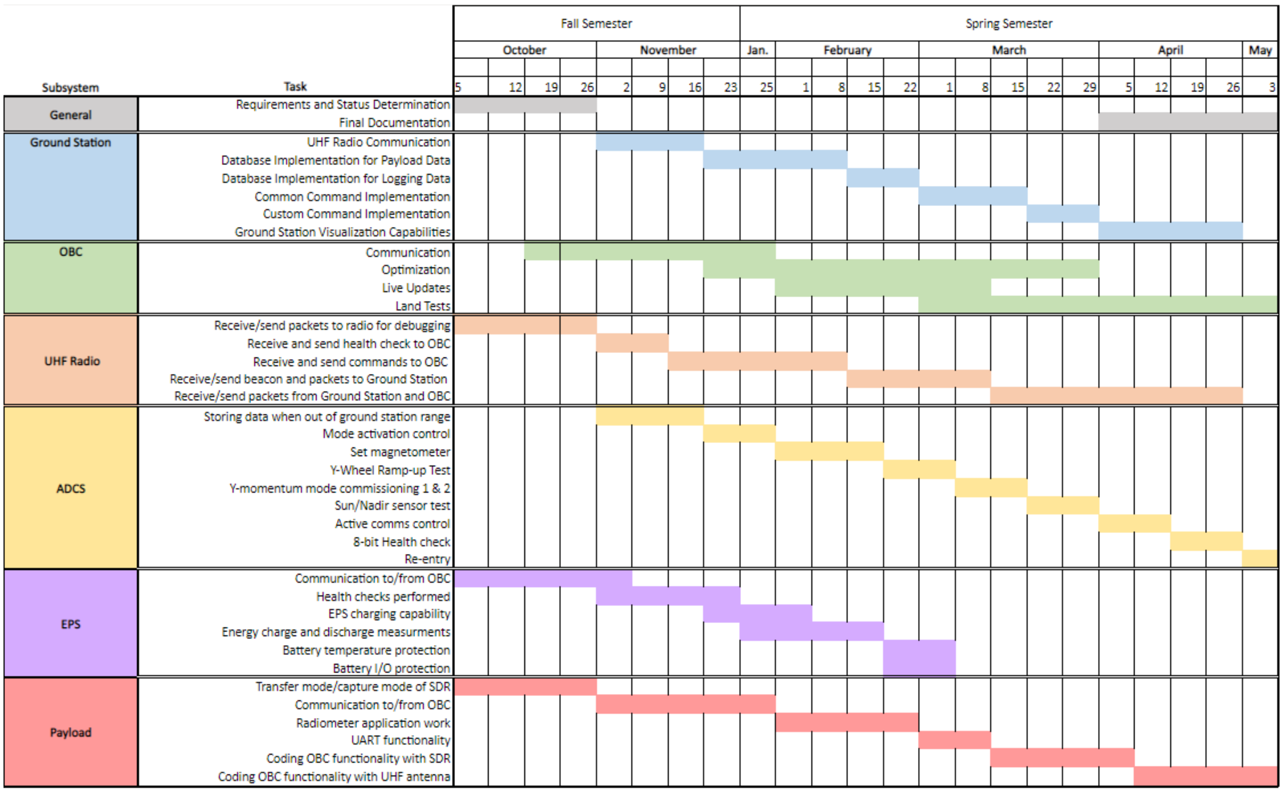
Milestone: OBC Mock Test Launch/Mission

Metrics: All subsystems will successfully communicate with the OBC. The OBC will receive commands from the Ground Station via the UHF radio. The OBC is in full control of what each of the subsystems are supposed to do.

Evaluation Criteria: All subsystems will perform as desired in space.

2.4 Project Timeline/Schedule

Our project timeline is grouped according to subsystem. This timeline delineates tentative due dates for numbered tasks from section 2.1. As we are using a SCRUM environment with week-long sprints, we afford ourselves the ability to change course and add/subtract both tasks and subtasks while maintaining a structured approach to project completion. Below is the initial, tentative breakdown of tasks:



2.5 Project Tracking Procedures

Our team will use Gitlab for version control. We will use Gitlab's issue boards to keep track of tasks and their relevant commits and branches, and to build and maintain a backlog of tasks. We will also be using slack for scheduling meetings, communicating about tasks, and asking questions.

2.6 Personnel Effort Requirements

Our tentative personnel effort requirements are laid out below, based on our initial understanding of project requirements and knowledge of the previous CySat teams' committed hours:

Substem	Task	Description	Hours
Ground Station	UHF Radio Communication	Sending, receiving, interpreting, and responding to CySat data	20
	Database Implementation for Payload Data	Permanent storage of Payload data, stored chronologically	25
	Database Implementation for Logging Data	Permanent storage of Commands Sent/Received and other Logs	15
	Common Command Implementation	Implementation of common commands to be sent to CySat	20
	Custom Command Implementation	Implementation of custom user command creation and use	12
	Ground Station Visualization Capabilities	Visualization of Satellite/Subsystems	20
	--	Total hours for Ground Station	112
UHF Radio	Receive and send packets from computer into radio for debugging	Hello world, Packet structure, addtional functional	30
	Receive and send health check to OBC	Make sure UHF is running properly	5
	Receive and send commands to OBC	Prompts OBC to access data or communicate with other subsystems	25
	Receive and send beacon and packets to Ground Station	Line of communctation between the satellite and its users	25
	Receive and send packets from Ground Station and OBC	Integration between OBC and Ground Station communcation	25
	--	Total hours for UHF Radio	110
Payload	Transfer mode/capture mode of SDR	Sending/collecting data via the SDR	20
	Communication to/from OBC	Being able to send or receive commands, data, etc between SDR & OB(	25
	Radiometer application work	Getting application to run on embedded Linux start up	20
	UART functionality	UART communication testing and completion	10
	Coding OBC functionality with SDR	Programming the OBC to be able to command the SDR using UART	20
	Coding OBC functionality with UHF antenna	Programming the OBC to transfer data using the UHF antenna	20
		Total hours from Payload	115
ADCS	Storing Telemetry data when out of ground station range	Recording telemetry dat for when the CySat is within range	12
	Mode activation control	Major component of programing the ADCS, flow control for op modes	40
	Set magnetometer configuration	Comput magnetometer offset and sensitivity matrix	12
	Y-Wheel Ramp-up Test	Testing for Y-Wheel Ramp-up Test	12
	Y-momentum mode commissioning 1 & 2	2 stages for Y-momentum mode commissioning	12
	Sun/Nadir sensor test	Testing of sun sensor for determining position	12
	Active comms control	Ground station control operational modes	12
	8-bit Health check	Additional time for health check of the system	15
		Total hours for ADCS	127
EPS	Communication to/from OBC	Update I2C to new version	40
	Health check	New I2C protocols add more parameters to be checked	25
	Charging and discharging	Measure and calculate the energy v. time of the batteries	30
	Battery protection	Change operation based on the data from the health check	15
	--	Total hours for EPS	110
OBC	Communication to all Subsystems	Sending, receiving, interpreting, and responding to all other subsyster	30
	Optimization	Look into FreeRTOS and use interrupts for UART and I2C	30
	Live Updates	Create bootloader that allows for live patches	30
	Land Tests	Implement mock mock launch, mock launch, and mock mission	50
	--	Total hours for OBC	140

Hourly time commitments are based on our understanding of the complexities of each subsystem, as well as their current states, and our estimations of the amount of work still needed for their completion. These hourly estimates are subject to change as the project moves forward.

2.7 Other Resource Requirements

This project is in direct collaboration with M:2:I. The M:2:I team has/will assist us with access to appropriate hardware and lab time. They will also give us feedback on what needs to be done, and how it affects the overall satellite.

3 Design

3.1 Previous Work And Literature

This project was inherited from last year's senior design and was originally started by Iowa State's M:2:I, and thus, much of the design process has already been completed. Due to the project being several years old, the previous work is incredibly extensive and has been provided through Git and Box.

Citations below are references to likely sources of information for the design process as M:2:I designed the CySat satellite. The associated websites, which can be found in the references section for each cited number, contain detailed information about the initiatives, products, and documentation relevant to the design of the CySat.

CubeSats have been in use for about 15 years, and all are fairly uniform in terms of design and operation[1]. However, each one can be focused towards a different research goal. Commercial companies supply various components that are CubeSat compatible. In this case, several of the CySat's subsystems are from the company EnduroSat[2], and the CySat's research focus is the health of vegetation.

It is important to keep the shortcomings of all the previous CubeSats in mind, since a very large percentage of all CubeSats fail. The goal of last year's senior design team was 'to not have a paperweight in space,' and it is being followed this year as well [3].

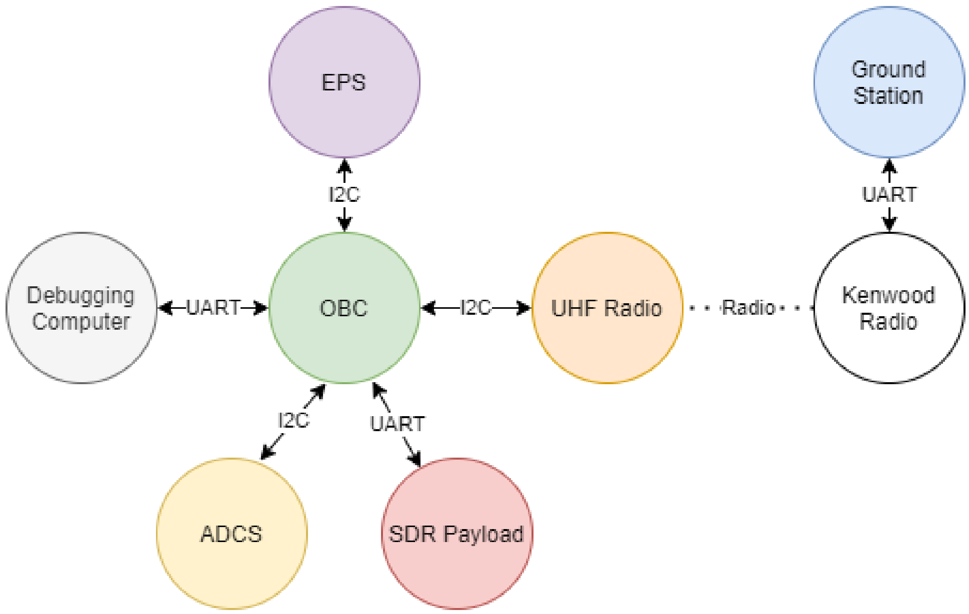
3.2 Design Thinking

There is little to no room for design thinking in this project. CySat is a 3U Cube Satellite, which have rigid design specifications that must be met in order for NASA to approve inclusion of the satellite into the payload. In addition, the CySat as a project has been in development since 2017 under of the supervision of the M:2:I community at Iowa State. The operational mission and internal hardware design of the satellite were specified long before our senior design team took on this project, and specific modules were selected from EnduroSat products available for CubeSats for the purpose of this project.

As such, we are making no high-level design decisions. Instead, our team's focus is on the implementation of commands in the various subsystems to accomplish the goals listed for each subsystem in section 2.1 Our team's goal is implementation of previously determined M:2:I designs, which themselves were defined based on the CubeSat specifications and available EnduroSat components.

3.3 Proposed Design

A flowchart of the CySat module communication design is shown below. As indicated in earlier sections, each of the circles below are subsystems of the CySat satellite. Each of these modules, aside from the payload, are required for operation of the satellite. Decisions for communication protocols (I2C and UART) were made by M2I prior to our inclusion in the project, and were chosen in order to conform to CubeSat and NASA standards.



Specifically, the diagram above has already been decided upon by M:2:I before our team's involvement in the project, and as such is the final design. The design above demonstrates the communication between components, and these communication channels are the channels which allow for the implementation of the functional requirements listed in 1.4.1. Those requirements are reiterated below, and the relevant connections that are involved in the performance of the functionality:

Requirement	Relevant Connections	Explanation
Must power up after been deployed from the International Space Station	OBC/EPS	The OBC and EPS will communicate to ensure that the satellite has power and powers up after the required 30 minute waiting period after deployment from the ISS
Must stabilize and point itself towards earth	OBC/ADCS	The OBC will communicate with the ADCS to begin and maintain detumbling and orientation
Must take soil moisture readings from Earth via a microwave radiometer	OBC/SDR Payload	The SDR collects data, which the OBC will compile

Must transmit data back to the ground station in Ames, IA	OBC/UHF Radio Ground Station/Kenwood Radio/ UHF Radio	The OBC will send data to the Kenwood Radio through the UHF, which will be received by the Kenwood radio which will communicate that data to the Ground Station
Must collect data for its orbit life (6 months)	All connections	All connection on the satellite work towards the overall mission goal of the satellite
Must meet NASA's CubeSat standards and regulations	All connections	All connections have been specified by M:2:I to conform to these standards and regulations
Must receive and execute commands issued by the Ground Station	Ground Station / Kenwood Radio and OBC / UHF	The Ground Station will communicate through the UART connection to the Kenwood, which will communicate with the OBC through the UHF Radio
Must successfully deorbit at the end of its lifespan	Ground Station / Kenwood Radio, OBC / UHF, and OBC / ADCS	The ground station will signal an EOL beacon to the satellite through the connections described above, and the OBC will communicate the intention to begin deorbiting to the ADCS

The non-functional requirement for the Ground Station performance is related to implementation, rather than design.

The design decision not shown in the diagram above is the CySat Packet Protocol. This design decision is somewhat between a design and implementation decision, and will be detailed in the Implementation section of this document. The overview of the packet protocol is that it is a standardized packet size that allows for easy decomposing of packets to allow for routing of messages by the OBC.

3.4 Technology Considerations

STMicroelectronics came out with the STM32CubeIDE Sprint 2019. Previously, one would use their STM32Workbench Eclipse plug-in for debugging and running code on the Endurosat OBC and the STM32F429ZI Discovery Board. Last years' senior design team utilized the Workbench plugin, but our team decided to move forward with using the STM32CubeIDE. The STM32CubeIDE allows for faster debugging (i.e. stepping over, stepping into, etc. are faster on the IDE). The STM32CubeIDE also allows for easy integration between the STM32CubeMX which allows for easy initialization of the different modules and associated interrupts (UART, I2C, etc.)

3.5 Design Analysis

M2I was responsible for this design, and they've given us and a previous group this design to implement. Because this is not our design, there is little analysis to perform, and so we note here that should a design level difficulty occur, we will make this difficulty clear to M2I and work towards a solution.

3.6 Development Process

We will be using a shortened Agile process with week-long sprints, as it allows for consistent parallel development of interrelated subsystems, and it will allow for the weekly meeting to serve as a sort of product demonstration. In addition, it will allow for a weekly check in, during Sprint Review, to ensure that all subsystems retain previously implemented functionality.

3.7 Design Plan

The proposed design from section 3.3 is the design plan for the CySat project, as the design was decided upon by M:2:I prior to our involvement in the project. Its relationships to functional requirements are shown in the table in section 3.3, and will not be reiterated here in the interest of space.

The subsystems will be implemented in parallel over the coming months, with frequent checks to confirm and reconfirm the continued performance of previously implemented functionality.

4 Testing

4.1 Unit Testing

Each subsystem must perform as expected; the I2C/UART interface for each subsystem is tested in isolation. Each subsystem must be able to receive a command from I2C and/or UART, perform the expected task properly, and return the proper response. Additionally, these tests must be done manually as there is no way to automate these tests.

4.2 Interface Testing

Our main testing will involve connecting the OBC, EPS, ADCS, SDR, UHF Radio, and Ground Station. The EPS, ADCS, SDR, and UHF Radio must be able to properly communicate with the OBC. They will receive commands from the OBC, and they must perform the required task properly. Additionally, the OBC must be able to schedule tasks for each of these subsystems. The UHF radio will be relaying commands between the OBC and the Ground Station.

The first test we will be running is called the "Mock" mock launch. This tests the communication between the OBC, the UHF Radio, and the Ground Station. The steps for this test launch are as follows:

1. Connect to Handheld Remote Machine(Ground Station)
2. Connect to OBC/UHF Remote Machine(Satellite)
3. On the satellite, power the pumpkin board with the power supply
4. Flash the OBC with the most current software
5. Power off the power supply
6. Open the Ground Station application on the Ground Station computer
7. Connect to the Handheld radio on the Ground Station

8. Power on the power supply and the following events should be observed automatically:
    a. OBC is powered on
    b. OBC simulates from handoff to NASA until power on of OBC
        i. Prints "RBF inserted"
        ii. Prints "Handoff to NASA...Loading into P-Pod"
        iii. Prints "Kill switch depressed"
        iv. Prints "RBF removed"
        v. Prints "Kill switch released (EPS power on)"
        vi. Prints "Entering Main()"
    c. OBC starts a 1 minute timer and waits, simulating the 30 minute period of inactivity following deployment from the International Space Station
    d. OBC turns on other modules in Satellite
        i. Simulates turning on UHF transceiver by printing "Commanding EPS to enable Output 5 (UHF transceiver)"
        ii. Simulates turning on SDR by printing "Commanding EPS to enable Output 3 (SDR)"
        iii. Simulates turning on Boost board by printing "Commanding EPS to enable Output 1 (boost enable)"
    e. OBC simulates deployment of magnetometer by printing "Commanding ??? to deploy the magnetometer"
    f. OBC simulates deployment of antenna by printing "Sending 0x1F (deploy all antennas with algorithm 1) to I2C slave address 0x33 (antenna)"
        i. Optional: Read the states of the UHF antenna and then deploy via algorithm 2 if needed
    g. OBC will send commands to the UHF Radio to configure the beacon text and period "Hello, Earth! I am ISU's CySat-I" with period of 1 minute
    h. OBC will send command to the UHF Radio to enable transparent mode
    i. OBC will send command to the UHF Radio to enable the beacon
    j. OBC will simulate beginning the detumbling sequence by printing "Beginning detumbling sequence"
    k. At some point after step h, the Ground Station will receive the beacon being sent by the Satellite. At this point the beacon should send a "Beacon Shut Off request" command as described in the Ground Station | Cysat Packet Protocol documentation.
    l. At some point after step k, the OBC will receive the "Beacon Shut off request" command
    m. OBC will turn off the beacon
    n. OBC will send a "Beacon shut off response" command
    o. Ground Station sends an "Initial Health Check Request" command
    p. OBC will run initial health checks (8-bit good/bad flags for each subsystem) and compile them into a response
        i. The EPS, ADCS, and SDR will simulate their health checks; we will assume they are all good.
        ii. The OBC and UHF Radio will run their actual health checks
        iii. OBC will send packet to Ground Station
    q. Ground Station will receive initial health checks and display them to the user.
    r. Ground Station will send a "Enter Main Operating Phase" request
    s. OBC will receive this and send a "Enter Main Operating Phase" response packet back to the Ground Station
    t. OBC will enter the main loop
        i. OBC will print "Main operating loop entered"

After this test passes, we will be stop simulating the EPS, ADCS, and the SDR payload, and these systems will be added to the tests. These more complex tests still need to be constructed along with expected results. In the end, we will form a mock launch and a mock mission with all subsystems communicating with each other as we would expect in space.

4.3 Acceptance Testing

We will be reviewing the results of the "Mock" mock launch, mock launch, and mock mission. We will make sure the results are as expected. We will also be performing testing with M:2:I to make sure that everything is performing as they would like.

4.4 Results

We have completed unit tests for the Ground Station and the OBC. We have made sure that each is able to communicate to properly over UART.

We have not been able to do any extensive testing due to the current state of this semester. We simply don't have remote access to all the subsystems yet.

5 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3.

6 Closing Material

6.1 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

6.2 References

[1] Jackson, S. (2017, February 17). NASA's CubeSat Launch Initiative. Retrieved October 26, 2020, from https://www.nasa.gov/directorates/heo/home/CubeSats_initiative

[2] EnduroSat - Class-leading CubeSat Modules, NanoSats & Space Services. (n.d.). Retrieved October 26, 2020, from https://www.endurosat.com/?v=7516fd43adaa

[3] Iowa State's CubeSat Project. (n.d.). Retrieved October 26, 2020, from https://m2i.aere.iastate.edu/cysat/

6.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.