CySat

DESIGN DOCUMENT

Team Number: 49
Client: Dr. Jones / M:2:I
Advisor: Dr. Jones
Bryan Friestad - Team Lead / EPS
Kyle Muehlenthaler - Radio
Ryan Hansen - SDR / Payload
Talon Stromgren - Analog Payload
Chase Kirchner - Ground Station
Xiangzhu Yan - ADCS

Team Email: sdmay20-49@iastate.edu
Team Website: sdmay20.ece.iastate.edu

Revised: 10/6/2019 (v1)

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

Summary of Requirements

List all requirements as bullet points in brief.

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

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

1 Introduction 4

Acknowledgement 4

Problem and Project Statement 4

Operational Environment 4

Requirements 4

Intended Users and Uses 4

Assumptions and Limitations 4

Expected End Product and Deliverables 5

2. Specifications and Analysis 5

Proposed Design 6

Design Analysis 6

Development Process 6

Design Plan 7

3. Statement of Work 7

3.1 Previous Work And Literature 8

3.2 Technology Considerations 8

3.3 Task Decomposition 8

3.4 Possible Risks And Risk Management 8

3.5 Project Proposed Milestones and Evaluation Criteria 8

3.6 Project Tracking Procedures 9

3.7 Expected Results and Validation 9

4. Project Timeline, Estimated Resources, and Challenges 9

4.1 Project Timeline 10

4.2 Feasibility Assessment 11

4.3 Personnel Effort Requirements 11

4.4 Other Resource Requirements **Error! Bookmark not defined.**

4.5 Financial Requirements 11

5. Testing and Implementation 12
Interface Specifications 12
Hardware and software 12
Functional Testing 12
Non-Functional Testing 12
Process 12
Results 13
6. Closing Material 13
6.1 Conclusion 14
6.2 References 14
6.3 Appendices 14

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

- M2I - Make 2 Innovate
- OBC - On board computer
- EPS - Electrical power system
- ADCS - Attitude determination and control system
- SDR - Software-defined Radio
- LNA - Low noise amplifier
- FPGA - Field programmable gate array

1 Introduction

1.1 ACKNOWLEDGEMENT

Significant funds have been provided for this project by the Iowa Space Grant Consortium, Make 2 Innovate and NASA, as well as other donors. The Make 2 Innovate team will also provide expertise and labor on other physical aspects of the project.

1.2 PROBLEM AND PROJECT STATEMENT

Make 2 Innovate (abbreviated M2I), a group on the Iowa State University campus has been in the process of developing a CubeSat to be put into orbit of Earth. A CubeSat is a small, uniformly sized satellite which will contain a scientific or research component. M2I has enlisted the help of our senior design team to write software for, test, and integrate the various components of the CubeSat (named CySat). In order to get the satellite into orbit, we have to complete these tasks by a deadline in the early spring, so we can hand the satellite off to NASA, who will facilitate the launch.

1.3 OPERATIONAL ENVIRONMENT

The end product will have to endure the intense vibrations of taking off from Earth in one of the International Space Station(ISS) resupply trips. From there, the satellite will be released from the airlock into Low Earth Orbit (LEO) where it will communicate the data it collects with a ground station in Ames, Iowa. These constraints impose great physical requirements that our team will not deal with very much.

1.4 REQUIREMENTS

- The ground station in Ames, Iowa must be functional and able to communicate with the satellite
- The satellite must not begin communicating with the ground station until 30 minutes after it has been released from the ISS
- The EPS must provide power to the other subsystems of the Satellite
- The OBC must delegate processing time to the various subsystems to insure all necessary tasks are completed in a timely manner
- The ADCS must be able to orient the satellite

1.5 INTENDED USERS AND USES

The end user of this product is the M2I group and the members of it who will collect and use the data transmitted by the satellite. It will be equipped with a radiometer, which will collect data about the water levels on Earth.

1.6 ASSUMPTIONS AND LIMITATIONS

Assumptions

- The satellite will deorbit and burn up after roughly 6 months in LEO
- If the satellite can pass our vibration tests, then it will withstand the launch to the ISS

Limitations

- The physical dimensions of the device are set to 3u(one u is 10cm x 10cm x 11.35cm)
- The satellite must not weigh more than 4kg (1.33 kg per u)
- We have a strict deadline in order to go up on the CRS2 NG14 supply launch

1.7 EXPECTED END PRODUCT AND DELIVERABLES

- OBC “hello world” program - Due ASAP
- SDR carrier board printed - Due ASAP
- LNA board designed and made
- OBC communicating with EPS, ADCS, SDR and UHF antenna
- Ground Station UI
- Ground Station communicating with OBC
- Dry fits 1, 2, and 3 (done by M2I team)
    - Dry fit 1 is physical constraint testing
    - Dry fit 2 is wiring
    - Dry fit 3 includes power up sequence
- Assembled Satellite - due early spring 2020

2. Specifications and Analysis

2.1 PROPOSED DESIGN

The functional requirement of this project is to design, develop, integrate, and test the software and electronic subsystems for CubeSAT. Key aspects of this mission the team will be responsible for are:

1) Integration of Commercial Off The Shelf (COTS) components into the Satellite structure.

2) Expansion of and integration with a Base Ground Station. The Base Ground Station will act as the mission control of the Satellite and will be housed in Howe Hall (i.e. in Aerospace Engineering Department). There already exists a ground station for communicating with High Altitude Balloon Experiments, and this will be leveraged as a starting point for completing a Base Ground Station for use with your Satellite.

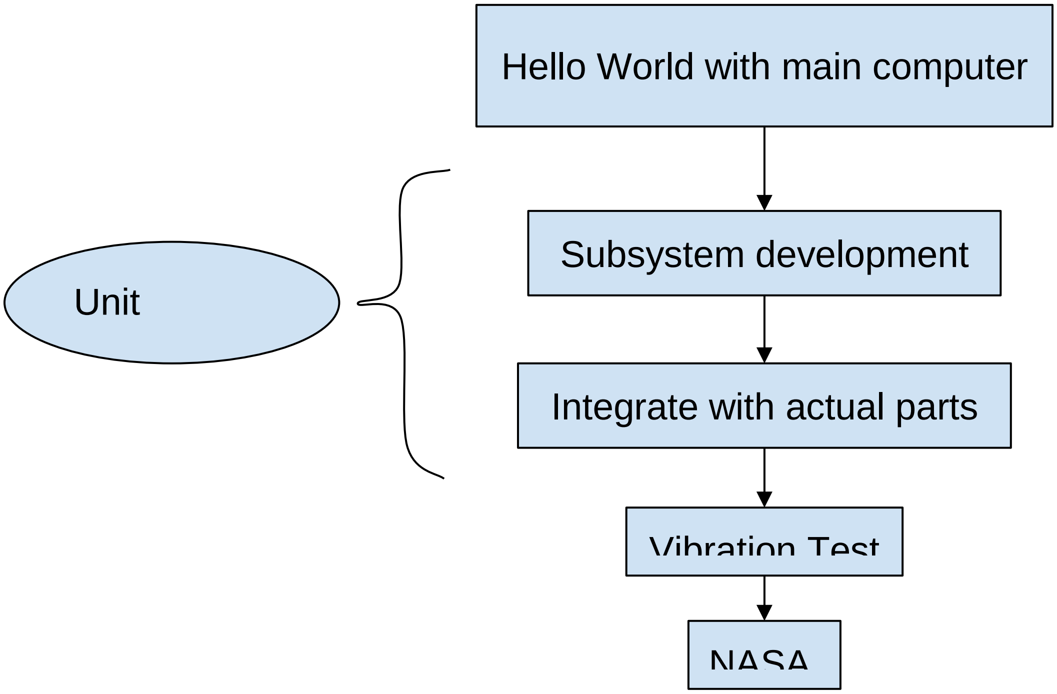
3) Scientific Payload Design and Development: In addition to integrating components of the Scientific Payload, there will be opportunities for innovation of the existing base functionality. This will involve learning about Software Define Radios (SDR), Embedded Linux, and potentially developing your own FPGA-based hardware components.

2.2 DESIGN ANALYSIS

We assigned individual tasks and responsibilities, we also set up the SDK for On-Board Computer and connect the Pumpkin board to our laptops by STLINK. So far all the work is going well (though there have been some difficulties). Since we've divided this project into several sub-systems and assigned them to group members, we will refer to the waterfall model we made to continue our work(our waterfall model is shown in the next section). The key result for is project is to integrate the sub-system we developed with actual parts of CySat.

2.3 DEVELOPMENT PROCESS

We use Waterfall model for our development process:



2.4 DESIGN PLAN

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

3. Statement of Work

3.1 PREVIOUS WORK AND LITERATURE

Iowa State University's M:2:I team has been working on developing a CubeSat for many years now. Over each semester they have optimized the design and chosen the hardware that would meet their mission needs. Our task as the software team is to work off the documentation and designs that have been established already and integrate the satellites various electronic systems.

In 2017, NASA published *CubeSat 101: Basic Concepts and Processes for First-Time CubeSat Developers* that details a lot of basic information of CubeSat development. A lot of our design implementation will be based off of the documentation in the CySat CyBox that has already been created for CySat 1.0 by the M:2:I team. We will be using *CubeSat 101* as well as the M:2:I documentation as our referenced literature for understanding the CubeSat architecture.

3.2 TECHNOLOGY CONSIDERATIONS

CubeSats use the PC104 standard for compact bussing of information signals around the satellite. Because this standard is widely used, finding standard components to interface with the PC104 stack is not difficult. Because the CubeSat will be put in space out of reach of human interaction after launch, components must be of good quality and pass many test cases which ultimately drives up the cost of the components.

Some hardware needed for the CubeSat are not available off the shelf in a suitable footprint or do not come with a PC104 connection. Custom PCB boards with the suitable dimensions and integrated hardware are required to fulfill some of the hardware tasks.

3.3 TASK DECOMPOSITION

CySat has various components that will all need to be worked on separately before the entire system is integrated. Communication with the OBC (On Board Computer) will be the first task that will need to be completed. From there we will need to establish communication from the OBC to the Radio Antena, SDR (Software Defined Radio), ADCS (Attitude Determination and Control System), and EPS (Electrical Power System). After communication with each major component is established sub level development and testing will begin.

3.4 POSSIBLE RISKS AND RISK MANAGEMENT

Our team has very basic knowledge of embedded systems and more specialized systems like that used on CySat will require learned knowledge and understanding. Unfamiliarity with certain electronics and their communication process may lead to troubles getting things up and working initially and this could potentially lead to delays in the development process.

3.5 PROJECT PROPOSED MILESTONES AND EVALUATION CRITERIA

Startup and "Hello World" with the OBC will be the first and most important milestone. Following the OBC startup, startup and communication with each of the 8 submodule components will be another important milestone. Final test fitting in CubeSat footprint and component integration with unit test completion will be the last milestone. Verification of each component's operation will be crucial before sending CubeSat to space.

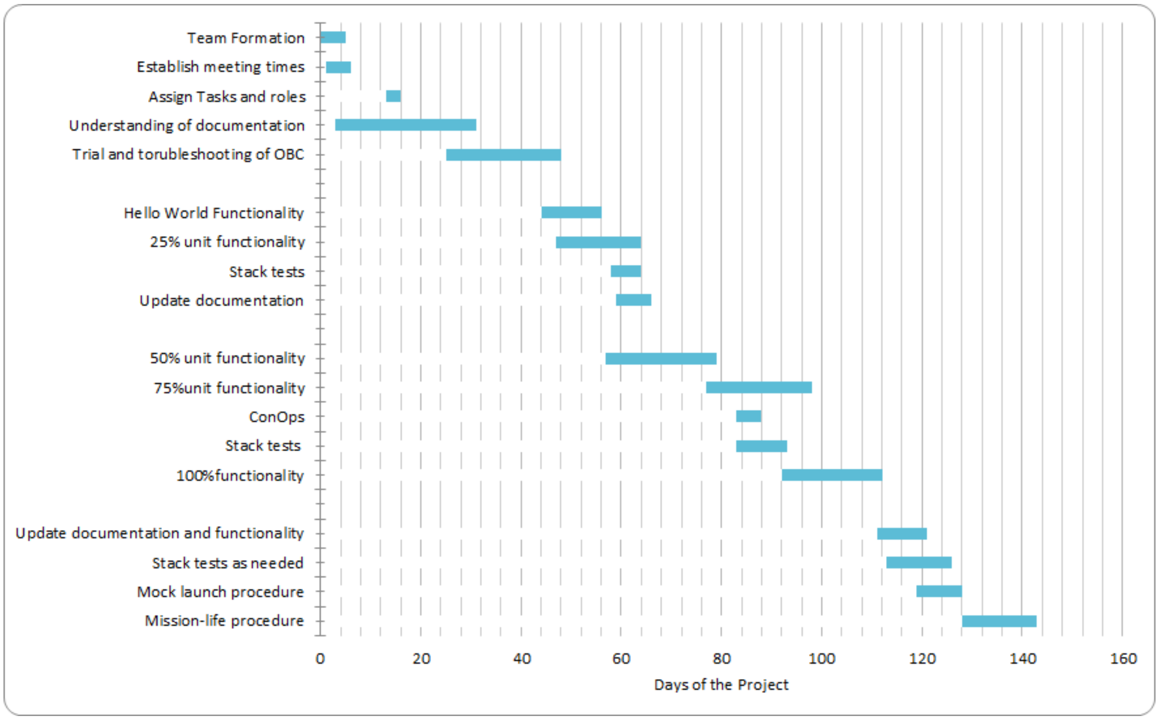
3.6 PROJECT TRACKING PROCEDURES

Progress will be tracked by noting the completion of each subsystem submodule objectives laid out by the M:2:I team and through weekly meetings with our client.

3.7 EXPECTED RESULTS AND VALIDATION

At the completion of this project we should have a fully operational and tested CubeSat that will be ready to launch into space to begin its mission. Initial communication and data retrieval with CySat after launch will be the testament of a successful design.

4. Project Timeline, Estimated Resources, and Challenges



4.1 PROJECT TIMELINE

August

- Team formed
- Established meeting times
- Assign Tasks and roles

September

- Achieve understanding of our roles
- Understand documentation
- Trial of equipment and trouble-shooting

October

- Hello world functionality among out subsystems
- Update of documentation
- 0-25% unit functionality
- Stack tests as needed

November

- 25-50% unit functionality
- ConOps discussion (incremental mission tasks)
- 50-75% unit functionality

- Stack tests as needed
- Update functionality

December

- 75-100% unit functionality
- Stack tests as needed, more in depth
- Begin mock launch procedure
- Update documentation

January

- Mock launch procedure
- Mission-life procedure

This timeline is compressed to make sure we make it to the January handoff to NASA but as a backup plan we also have a June handoff that will end up being more likely. This is achievable, however we will be pressed and may change course, this will be updated to reflect that as decisions are met.

4.2 FEASIBILITY ASSESSMENT

This project has gone through many tough parts as we have gone through sickness and troubles with our OBC, this is still a feasible project, however we are going to need to find time to come together as a single team and knock out some issues we are having in our phases.

4.3 PERSONNEL EFFORT REQUIREMENTS

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be based on the projected effort required to perform the task correctly and not just “X” hours per week for the number of weeks that the task is active

4.4 FINANCIAL REQUIREMENTS

All funding has been provided by M:2:I and hardware has already been acquired. Hardware is estimated at over $120,000.

5. Testing and Implementation

NASA requires extensive testing in order to deploy a satellite into space. Testing requirements include functional, non-functional, vibe, and launch testing to name a few.

Testing Plan:

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study for functional and non-functional requirements)
2. Define the individual items to be tested
3. Define, design, and develop the actual test cases
4. Determine the anticipated test results for each test case
5. Perform the actual tests
6. Evaluate the actual test results
7. Make the necessary changes to the product being tested
8. Perform any necessary retesting
9. Document the entire testing process and its results

5.1 INTERFACE SPECIFICATIONS

Subsystems will interface with OBC throughout satellite path. OBC will be interfaced to Ground Control for data collection and satellite status checks.

5.2 HARDWARE AND SOFTWARE

Vibe Testing: Requires specialized chamber to verify physical components

Regression Testing: Will be completed through automated and manual procedures

-Automate testing processes within Gitlab after deployments are made

– Indicate any hardware and/or software used in the testing phase

– Provide brief, simple introductions for each to explain the usefulness of each

5.3 FUNCTIONAL TESTING

Examples include unit, integration, system, and acceptance testing

- Unit testing for each subsystem - Making sure each subsystem is fully functioning and communicating properly.
- Vibration testing - Making sure all physical components will stay pieced together on the space shuttle

5.4 NON-FUNCTIONAL TESTING

Testing for performance, security, usability, compatibility

5.5 PROCESS

– Explain how each method indicated in Section 2 was tested

– Flow diagram of the process if applicable (should be for most projects)

5.6 RESULTS

– List and explain any and all results obtained so far during the testing phase

- – Include failures and successes
- – Explain what you learned and how you are planning to change it as you progress with your project
- – If you are including figures, please include captions and cite it in the text

  • This part will likely need to be refined in your 492 semester where the majority of the implementation and testing work will take place

-**Modeling and Simulation**: This could be logic analyzation, waveform outputs, block testing. 3D model renders, modeling graphs.

-List the **implementation Issues and Challenges**.

6. Closing Material

6.1 CONCLUSION

As of October 6, 2019, we have gone through a rigorous process of trying to understand and organize all of the documentation for the project and its subsystems. Our goals include getting communication with the On-Board Computer, getting all subsystems to communicate with each other, and achieving thorough unit testing functionality. Our plan of action is to install the basic software needed on a few different laptops so we can try different ways of communicating with the OBC as well as meet every time during our allotted work blocks. After we have reached that goal, we will dive into the heavy development stage and writing unit tests at the same time. This is the best way to achieve our goals, because everyone will be able to bounce ideas off of each other and we will be spending a lot of time in the lab.

6.2 REFERENCES

Our references include all documentation that we have regarding the specific components of the satellite. These are manuals that are provided by the companies where we purchased the components.

6.3 APPENDICES

One large problem we encountered was that the specific files we need to work with the OBC are not version controlled well by Eclipse and it completely messes up the project if you try to import it. There also may be an issue with our JTAG to USB converter, but we have yet to test that.