NOAA GOES-R Satellite Receiver

Dr. Nathan Neihart
Nick Butts - Software Development
Yong Lim - Software Development
Jonathan Massner - Systems and RF Design
Theodore Mathews IV - RF Design
Riley Stuart - ADC Design
Jordan Tillotson - ADC Design

Team Email and Website:
sdmay20-03@iastate.edu
sdmay20-03.sd.ece.iastate.edu

December 12, 2019

Executive Summary

Development Standards and Practices Used

- Agile Development
- IEEE 145-2013: IEEE Standard for Definitions of Terms for Antennas
- IEEE 149-1979: IEEE Standard Test Procedures for Antennas
- IEEE Standard 211-2018: IEEE Standard Definitions of Terms for Radio Wave Propagation

Summary of Requirements

Hardware

- **Antenna:** WiFi grid antenna with a design frequency of 2.4 GHz.
  - Size Requirement: 1.1 meters in diameter
  - Frequency: 1694.1 MHz. The antenna will be modified for this frequency.
  - Connector: Female N
  - Polarization: Vertically Polarized
  - Aiming Accuracy: ±5° on all axis
- **Filtering & Amplification:** Following the Antenna, the signal needs to be filtered and amplified.
  - Gain: > 30 dB
  - Bandwidth: 2MHz (Signal is 1.205MHz)

Environmental & Physical

- **Line of Sight:**
  - The antenna needs a clear line of sight to the satellite, limiting possible setup locations.
- **Semi-Portable:**
  - Unable to permanently mount the antenna on campus due to cost and planning requirements
  - Long reception times complicate testing
- **Clouds & Weather:**
  - Cloud cover should not impact reception.
  - Equipment will need to be weather resistant if a permanent installation is desired.
- **Budget:**
  - $\approx$ $1,000
- **Components**
  - Use COTS (Commercial off-the-shelf) where possible for larger components (antenna, preamps, computers).
  - Radio Frequency (RF) and Analog-to-Digital (ADC) sections are the primary hardware design foci.
- **User Interface (UI):**
  - The Raspberry Pi 4 (RPi4) we are using will host a website that displays information about the download process and completed images.

Applicable Courses from Iowa State University Curriculum

- EE 414: Microwave Engineering
- EE 230: Electronic Circuits and Systems
- EE 333: Electronic Systems Design
- EE 224: Signals and Systems I
- EE 324: Signals and Systems II
- EE 321: Communication Systems I

New Skills/Knowledge Acquired

- All members of our team are Electrical Engineering majors, so we do not have the training that Software Engineering majors would have. We still have a strong programming background from varying from individual experiences.
- The use of GitHub for code management and general coding guidelines
- Detailed design and simulation of RF systems
- Advanced topics in Applied Digital Communications

Contents

1 Introduction 5

1.1 Acknowledgement 5

1.2 Problem and Project Statement 5

1.3 Operational Environment 5

1.4 Requirements 5

1.5 Intended Users and Uses 6

1.6 Assumptions and Limitations 6

1.7 Expected End Product and Deliverables 7

2 Specifications and Analysis 7

2.1 Proposed Design 7

2.2 Design Analysis 8

2.3 Development Process 8

2.4 Design Plan 10

3 Statement of Work 11

3.1 Previous work and Literature 11

3.2 Technology Considerations 11

3.3 Task Decomposition 12

3.4 Possible Risks and Risk Management 12

3.5 Project Proposed Milestones and Evaluation Criteria 13

3.6 Project Tracking Procedures 13

3.7 Expected results and validation 13

4 Project Timeline, Estimated Resources, and Challenges 14

4.1 Project Timeline 14

4.2 Feasibility Assessment 16

4.3 Personnel Effort Requirements 16

4.4 Other Resource Requirements 18

4.5 Financial Requirements 18

5 Testing and Implementation 18

5.1 Interface Specifications 18

5.2 Hardware and Software 18

5.3 Functional Testing 19

5.4 Non-Functional Testing 19

5.5 Process 19

5.6 Results 19

6 Closing Material 19

6.1 Conclusion 19

6.2 References 19

6.3 Appendices 20

List of Figures

1 Team Overview 9

2 High Level System Overview 10

3 LNB Details 10

4 Receiver and ADC 10

5 Project Gantt Chart 15

List of Tables

1 Personnel Effort Requirements 17

Definitions

- μC: Microcontroller
- RPi4: Raspberry Pi 4
- IDE: Integrated Development Environment
- LNA: Low Noise Amplifier
- UART: Universal Asynchronous Receiver-Transmitter
- DSP: Digital Signal Processing
- NOAA: National Oceanic and Atmospheric Administration
- OS: Operating System
- PLL: Phase Locked Loop
- IF: Intermediate Frequency
- GOES: Geostationary Operational Environmental Satellite
- ADC: Analog to Digital Converter
- SPI: Serial Peripheral Interface
- SDR: Software Defined Radio
- RF: Radio Frequency
- ETG: Electronics Technology Group
- COTS: Commercial Off The Shelf
- UI: User Interface
- PCB: Printed Circuit Board
- FPGA: Field Programmable Gate Array
- LNB: Low Noise Block down converter
- IEEE: Institute of Electrical and Electronics Engineers
- USB: Universial Serial Bus
- ISM: Industrial Scientific and Medical
- AC: Alternating Current
- PC: Personal Computer
- GPIO: General Purpose Input/Output
- RAM: Random-access Memory

1 Introduction

1.1 Acknowledgement

Dr. Nathan Neihart, an Associate Professor in Electrical Engineering at Iowa State, will be the faculty advisor for this project. He is providing technical support and teaching important fundamental principles needed for our project. The ETG, located at Iowa State University's College of Electrical and Computer Engineering, will also be providing support for the hardware portion of this project.

1.2 Problem and Project Statement

The overall goal of this project is to build a satellite receiver that will receive image products from the Geostationary Operational Environmental Satellite - 16. The satellite is in geostationary orbit and sends earth image data that is accessible to anyone. In order to successfully retrieve the image, the end user must design a system to receive and process data from the satellite. The focus of this project is to design an entire end user system using commercial, off-the-shelf components, and document the process to give electronics hobbyists an idea of how to approach similar projects on their own.

To complete this project, the team must find a way to receive the RF signal from the satellite, extract the binary data, and process the data from the earth image. To do this, the group is divided into three sub-teams; RF, DSP, and software. The RF team is responsible for designing a method to receive the RF signal from the GOES-16 satellite and convert it to a signal that is capable of being processed by standard ADC components. The specifications for the RF portion will be tested in the on-campus RF lab. The DSP team is leading the effort of processing the signal produced by the RF team and extracting the binary data. This completed through the use of an ADC and a methodology of reading the digital signal. It will be tested in a lab environment using simulated signals and standard digital test equipment. The software team will be in charge of processing the binary data from the DSP team and producing an earth image.

1.3 Operational Environment

The operational environment for the final product will be a combination of a typical laboratory environment and an outdoor environment. The antenna must be outdoors long enough to receive one transfer of the image data coming from the satellite. All reception of data will occur on clear days when the weather is fair to prevent this project from requiring weatherproof components. The data from the satellite may be saved and processed in an indoors lab environment through software.

1.4 Requirements

- **Antenna:** The antenna for this receiver will be a Wifi grid antenna with a design frequency of 2.4 GHz.
  - Size Requirement: 1.1 meters in diameter
  - Frequency: 1694.1 MHz. The feedhorn must be modified for the desired frequency.
  - Connector: N or SMA
  - Polarization: Linear, vertical orientation
  - Aiming: ± 2° in both rotation and elevation to ensure maximum gain
- **Filtering & Amplification:** Following the Antenna, the signal needs to be filtered and amplified.
  - Gain: > 30 dB
  - Bandwidth: 2MHz centred at the above frequency.
- **Line of Sight:**
  - The antenna needs a clear line of sight to the satellite, limiting possible setup locations.

- **Semi-Portable:**
  - Unable to permanently mount the antenna on campus due to cost and planning requirements
  - Long reception times complicate testing
- **Clouds & Weather:**
  - Cloud cover should not impact reception.
  - Equipment will need to be weather resistant if a permanent installation is desired.
- **Budget:**
  - ≈ $1,000
- **Components**
  - Use COTS (Commercial off-the-shelf) where possible for larger components (antenna, preamps, computers).
  - RF and ADC sections are the primary hardware design foci.
- **UI:**
  - The RPi4 we are using will host a website that displays information about the download process and completed images.

1.5 Intended Users and Uses

This project is intended to provide the ability to receive weather products from NOAA via the GOES-16 weather satellite without needing a connection to the internet.

There are three intended users groups for the project.

1. **Dr. Neihart:** This project was proposed by Dr. Neihart to expand his ability to receive weather products from space.
2. **Hobbyists:** There is a large community of hobbyists around the world that seek to receive images and information from weather satellites. This project would provide them with an option that is tailored for the GOES-16 satellite.
3. **Emergency Responders:** There have been commercial releases of products that provide emergency coordinators with critical weather information “off the grid”. This project could provide them with a lower cost option to supplement any existing systems that they may have.

1.6 Assumptions and Limitations

Assumptions:

- The receiver will not be designed to be permanently mounted.
- End users have working knowledge and tools to correctly position the antenna.
- End users have access to computer and basic computer skills.
- The product will be used in an area where the GOES-16 satellite is visible.

Limitations:

- The signals the receiver can process will be limited to the GOES-16 and GOES-17 satellites.
- The receiver cannot be placed in inclement weather situations.
- The system requires AC power for the RPi4 and Receiver.
- The cost of the final product should not exceed $1,000.

1.7 Expected End Product and Deliverables

The end deliverable will be a complete receiver and data processing system. The finished system will be capable of receiving image data from a satellite, processing the data, and uploading the image to an easily accessible folder. The system will not require any user input once it has been setup and activated. To achieve this goal, there are a variety of sub-systems that must be delivered.

The first sub-system that must be delivered is the antenna and RF portion. To complete this system, an antenna capable of receiving the Wifi signal will be chosen, along with all filtering and amplification components necessary. The products delivered will include an antenna, cables and connectors, and a PCB containing all circuitry necessary to output a baseband signal.

The second sub-system that will be delivered is the signal processing components. This system will include all of the signal processing components necessary to extract the binary data from the baseband signal. The final deliverable for this portion of the system will be a PCB with connectors, an ADC, and a μC.

The last sub-system is the software deliverable, which will generate an image file from the binary data. This system will include an executable software file which will include all external libraries and references. The final deliverable for this portion of the project will be an executable software file that can be run on a RPi4, along with a file storage system to hold all received images.

The entire system will include all of the aforementioned sub-systems along with all cables, connectors, and power supplies necessary to integrate the final product. In addition to that, a setup and troubleshooting guide will be provided for users to refer to. All of the deliverables will be delivered by April 21, 2020.

2 Specifications and Analysis

2.1 Proposed Design

A variety of design approaches exist to implement the product described in the previous sections. Each of these methods vary in slightly different ways, but have a similar overall approach. In general, all methods of implementation include a way to receive the signal from the satellite, process the received RF signal, and develop an image from the processed information. The team has experimented with a few of these approaches thus far, and will narrow down the final design based on the successes and failures.

Receiving the signal is an important aspect of the final design. There is only one way to actually capture the signal from the satellite, which is through an antenna. The proposed antenna is a purchased antenna capable of receiving WiFi signals. The antenna should be adjusted to properly pick up the signal from the GOES-16 satellite. The next stages of RF signal reception are more flexible. So far, the team has experimented with a software defined radio (SDR) to observe the signal from the satellite. Using an SDR allows the signal to be directly sent to a computer to complete the remaining signal processing algorithms. This approach is very attractive due to most of the project being done on a computer, however, the team's goal is to create a hardware implementation of the signal processing. With that, the alternative is to design the hardware necessary to amplify, filter, and demodulate the RF signal. The team is currently working on the development of the RF hardware.

Processing the signal and extracting the binary data is the second stage of the final design. The first approach the team investigated was using an FPGA to do all of the desired digital signal processing. This method would require many hours of test and development, and given the lack of team experience with this implementation, it was decided that this option would not be the best. The second approach, which is the adopted approach, is to use an analog to digital converter (ADC) to read the demodulated signal, and perform computations on the data with a μC to determine the resulting binary waveform. This implementation will be slightly slower than the previous design approach, but the team has the necessary experience to implement it. The development of this approach is currently underway by some members of the team.

The final portion needed to implement the project design is software for generating the image file. The team investigated a variety of ways to implement the software processes by researching different languages, operating systems, and software libraries. The initial approach to implementing the software was to write mostly original software in C programming language and use our software to process digital image data delivered by the ADC team. Writing the software in C programming language would provide fast and efficient data processing. Through further research, the software team determined that the complexity level of C programming necessary for this project was too high for the time allotted. The decision to use external libraries greatly minimized development time and allowed more time to focus on processing image data. The initial design included processing the digital data on a PC but evolved into using a RPi4.

2.2 Design Analysis

The largest design difficulty with this project comes from the group's lack of experience in the area of satellite communications. The majority of the team did not have any prior knowledge of the GOES-16 satellite and the requirements for interfacing with the satellite. Due to this, we completed all research from the beginning of the engineering process. Through this research, multiple design ideas and examples were found.

The proposed design for the project as stated in *Section 2.1* holds both strengths and weaknesses. The RF front-end is one of the project's strengths, using both COTS components as well as a PCB to receive the signal and down-convert it to IF. One weakness of the proposed design was within the software implementation. It was observed that the difficulty in writing a decoding library was too great, and a revised approach has been implemented.

2.3 Development Process

The primary development process that is being utilized for the project is Agile in nature. This is being facilitated through the project's GitLab repository and its associated planning and management tools.

The project has been split into three main teams; RF, ADC, and Software. Each team will be responsible for managing its own development process. A systems leader will collaborate with each of the teams to ensure the group is meeting time and budget requirements. The resulting structure of the teams can be seen in Figure 1.

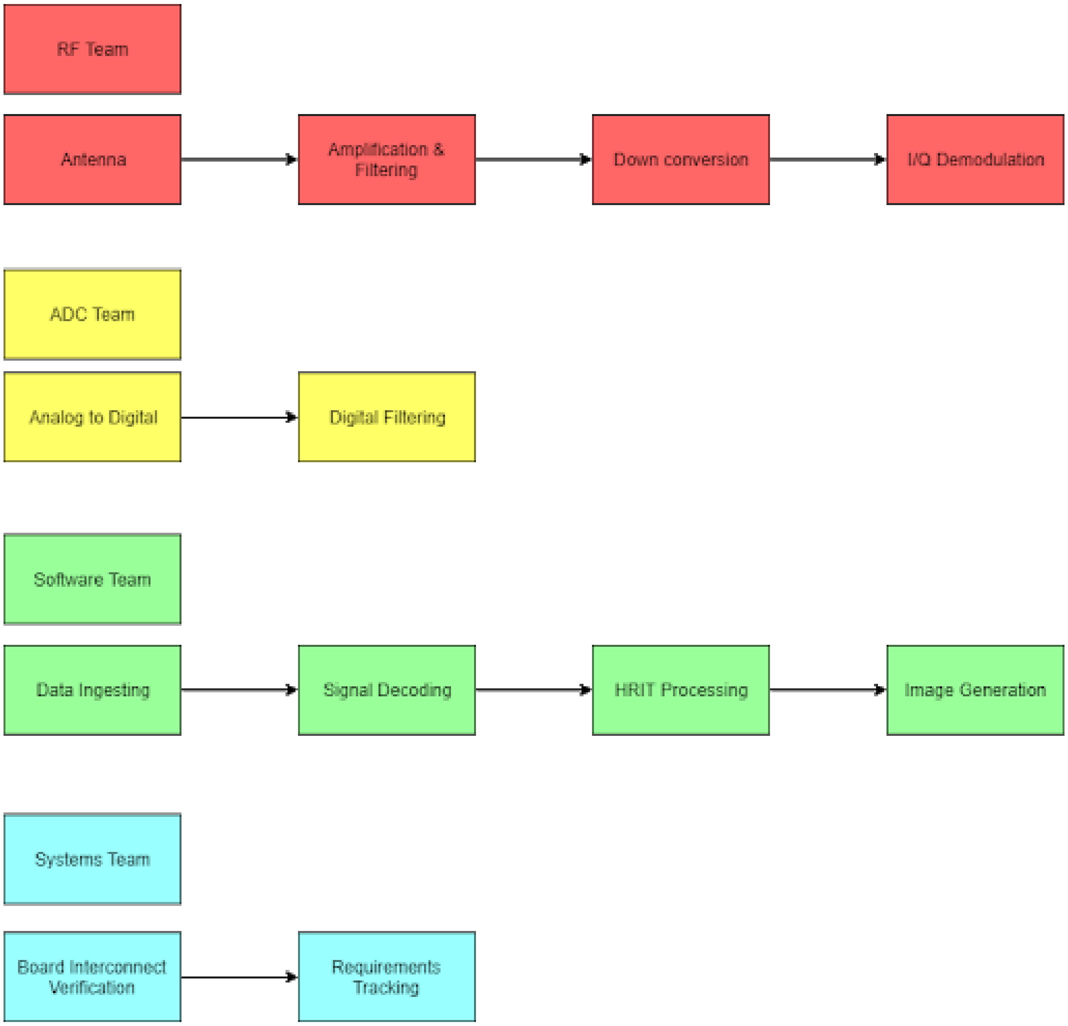

Figure 1: Team Overview

2.4 Design Plan

The overall design plan for the project is shown in Figure 2. The design plan includes an LNB down-converter shown in Figure 3, and and analog to digital converter shown in Figure 4. The team will follow this plan while designing the system.

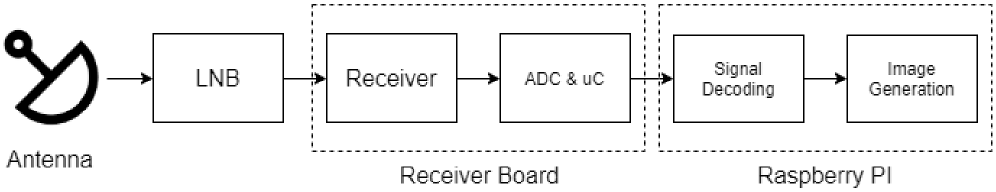

Figure 2: High Level System Overview

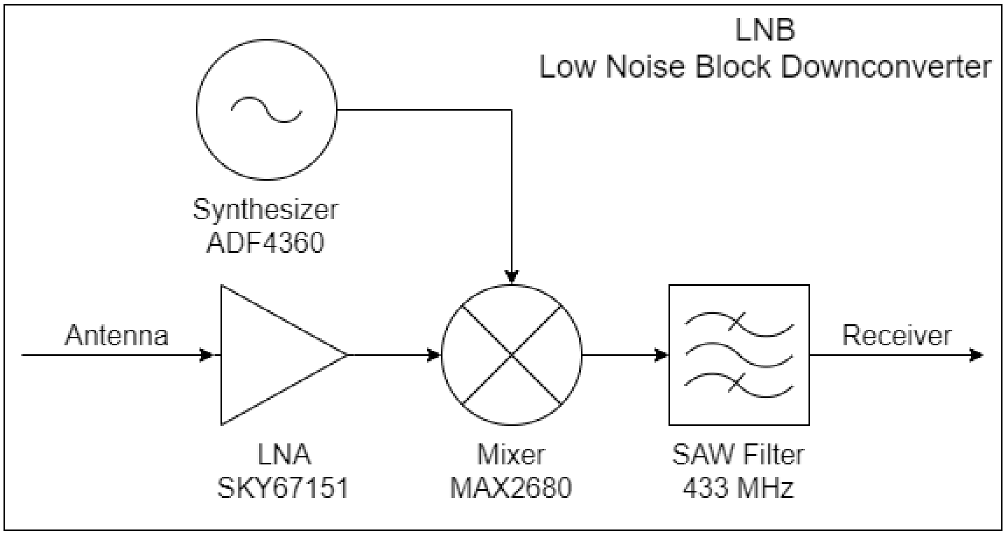

Figure 3: LNB Details

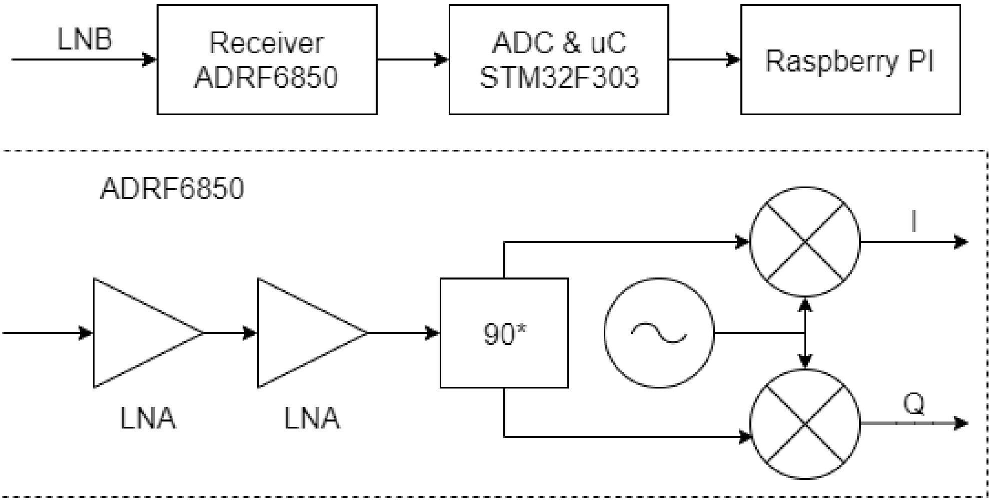

Figure 4: Receiver and ADC

3 Statement of Work

3.1 Previous work and Literature

There has already been extensive work on this type of receiver by both engineers and hobbyists. There is hardware and software available for purchase to easily implement the entire system; our team is attempting to design and implement our own system that does the same thing.

One project that we referenced (Roberson et al.) used a combination of RF hardware and an SDR to extract the information from the satellite signal. The antenna for the design was a 1.0 meter antenna which was connected to a Low Noise L-Band Block (LNB) downconverter. The IF signal from the LNB was then connected to a circuit board containing an ADC and FPGA. The ADC was used to digitize the IF signal, and the FPGA was programmed to handle all digital signal processing duties. The digital signal from the FPGA was then sent through USB to an SDR, which handled all of the signal receiving and checking. This particular system is close to what we want to do and we will use this as a guide.

In general, we will be modifying this system in a few ways. We will use more components in the RF signal manipulation stage and also try to use software to do everything the FPGA is doing. Using software to do this will simplify what the system is doing as a whole. However, the antenna used in the model system will likely be the same antenna we use to implement the system.

There is an open-source Github project, called the Open Satellite Project, made primarily by Lucas Teske (Brettbolt) that has the same goal as our team. We will be using the code he has written for the decoding of the signal and file generation for viewing. The Github repository can be found at the following link:

https://github.com/opensatelliteproject

The Open Satellite Project is a robust solution to the problems of demodulating and decoding the signal and reconstructing the image files. However, our team does not have the time to fully explore and understand the code. This makes troubleshooting the code difficult and time consuming and testing will potentially be problematic when we are ready to test the system.

With regards to background and literature, we are learning the necessary concepts through the use of books like Digital Communications (Sklar) as well as RF Microelectronics (Razavi).

3.2 Technology Considerations

RF Section

- Receiver architecture is a mix between an SDR and a superheterodyne receiver leveraging the strengths of each.
- Designing RF chain to take advantage of commonly available filters, specifically ISM band parts.
- Design RF systems for fabrication on FR4 based PCB's to control costs and allow for faster revisions.

ADC Section

- Data requirements from incoming IF signal (data rate, amplitude and offset, etc.)
- Capabilities of the ST32F303 μC (resolution, clock speed, voltage range, GPIO, etc.)
- Communication interface from μC to other components (RPi4, I/Q Demodulator, PLL)
- Software development platform (STMCubeMX IDE, ARM Mbed, PlatformIO)

Software Section

- Software must compile under Debian on a RPi4
- RAM must be large enough to be able to process image received in a reasonable time frame

3.3 Task Decomposition

The work is distributed among the teams we have assigned.

- The RF team is responsible for receiving the signal and demodulating it to allow for digitization.
- The ADC team will digitize the signal and transfer the data onto a RPi4 for image processing.
- The Software team will write the code to interpret the raw binary data, generate an image, and upload it to a website.

3.4 Possible Risks and Risk Management

Project Completion Risks:

- Understanding how to process data packets into images requires a substantial amount of background knowledge in digital signal processing. Prior to this project, the team's combined educational and professional experience with DSP was very limited.
- The inability for hardware and software matching to meet the speed of data transfer would prove detrimental to project completion.
- Signal strength must be boosted to meet processing requirements. If the signal strength is not strong enough, the rest of the system will have trouble processing the signal and will not produce an image.
- Risk to hardware exists by improper connection and powering of system components.

Physical and Environmental Risks:

- The physical and environmental risks for this project were low.
- Physical risk presents when improperly transporting equipment such as the antenna.

Risk Mitigation:

- The team tirelessly researched DSP techniques and practices to gain a general understanding for moving forward with the project.
- Thorough analysis of datasheets was performed to determine if system components abilities were sufficient.
- The use of multiple low noise amplifiers (LNAs) ensured the signal strength was strong enough to meet processing requirements.
- All system integration was performed meticulously to ensure no damage to components.
- Large hardware components were moved and assembled as a team to lessen physical strain.

3.5 Project Proposed Milestones and Evaluation Criteria

There are a number of major milestones that will be completed throughout the course of the project. Each of our sub-teams have milestones of their own, and the final system itself will have milestones.

For the RF team, the first major milestone to accomplish will be to successfully receive, amplify, filter, and demodulate the incoming signal. To do this, we will test each of the sub-components in the full assembly, and then integrate the components to form the final prototype. The first of these smaller milestones will be to receive and amplify the incoming signal. To test this, a spectrum analyzer will be used to observe the signal power and frequency bands being received by the antenna. We have successfully received the signal if we can clearly see a signal centered at 1694.1 MHz. For the next minor milestone, we will incorporate the filtering components into the design and perform testing with a spectrum analyzer to confirm that signals in nearby bands have been filtered out. Next, the downconversion components will be added and we will observe the baseband signal before moving on to the I/Q demodulation. To complete the milestone of successful demodulation, we will use a test board and all combined components to observe the I/Q demodulated signal on an oscilloscope. Other major milestones throughout the project will include final PCB design, and successful integration of the RF parts into the final design.

The first major milestone for the ADC team to accomplish will be to successfully communicate to all of the necessary components within the project. This means that the μC will correctly use SPI to configure the IQ Demodulator and PLL, as well as use UART to communicate to the RPi4. The second milestone for the ADC group will be to successfully receive and digitize the incoming signal. To do this, correct connection to the RF hardware will need to be ensured, as well as correct configuration of the ADC within the μC. To test these milestones, an oscilloscope can be used to verify the messages sent over SPI, and test data can be digitized and sent to the RPi4.

The Software team will be using the code from the OpenSatelliteProject(OpenSat), and the first key milestone would be to successfully compile the code available on the repository of the OpenSat. The first task to this milestone would be to first clone and push all the available code on the OpenSat repository to the group's Gitlab repository. The next task would be to install an OS (Raspbian) onto the micro SD card of the RPi4. To complete this milestone, the code will be compiled using specific commands listed in the README.md file available in OpenSat folder. (Will need to be more detail about this) Other major milestones include decoding the data received from the antenna and generating an image using the decoded data.

3.6 Project Tracking Procedures

This project will be tracked mostly through Git. The Wiki feature will be used to keep track of relevant information pertaining to the project, and allow us to explain and document research about the project for others to view. The project repository will be used for all software files, with multiple people contributing to the software throughout the development. Non-software files are kept on team members personal computers through development, and then uploaded to the repository when an individual task has been completed. The last major part of our project tracking procedures is using issues within Git to organize and keep track of past, present, and future tasks. Team members are responsible for creating stories as they deem necessary, and writing updates within the issue as they progress.

In addition to Git, the team has weekly meetings with Dr. Neihart where the group's progress is discussed and what areas need to be worked on for the following week. Additional meetings are held as a team, discussing how the different groups need to come together at the boundaries of their respective responsibilities.

3.7 Expected results and validation

The high-level end result for this project will be to successfully acquire Earth images through the satellite. The system outputs an image file which can be viewed and analyzed for image quality. After each of the sub-components have been validated through the process described above, we will test the integrated system to confirm correct signal and data transmission between our sub-systems. Finally, the image output will be observed to try to find areas of lower quality, which may be indicative of some faulty system components. If the image does not meet quality expectations, each sub-system will be broken down and tested once more to attempt to find the bug in the system.

4 Project Timeline, Estimated Resources, and Challenges

4.1 Project Timeline

The Gantt chart shown in Figure 5 outlines the general progression of the project throughout the course of the two semesters. The project has been broken down into four time periods of development; research, prototyping, testing, and finalizing the project.The tasks shown in the chart correlate with the requirements previously discussed. Overall, the team schedule accounts for individual sub-team progress, along with the overall progress for the team.

The project schedule has been created to complete the project in two semesters. To do this, the team has worked to have the project researched thoroughly and the requirements defined by the end of October of the first semester. As research is being conducted into the project, some of the sub-teams will begin some of the initial prototype work, which will include mostly software coding, and purchasing some development kits for chosen hardware. The initial prototypes for each of the sub-teams will be completed early in the second semester to allow a significant amount of time for testing and debugging. Most of the second semester has been set aside for troubleshooting and testing of the system, with time scheduled in the second half of the semester to develop the final product. The project will be completed in May of 2020 after all of the final hardware and software reviews are done.

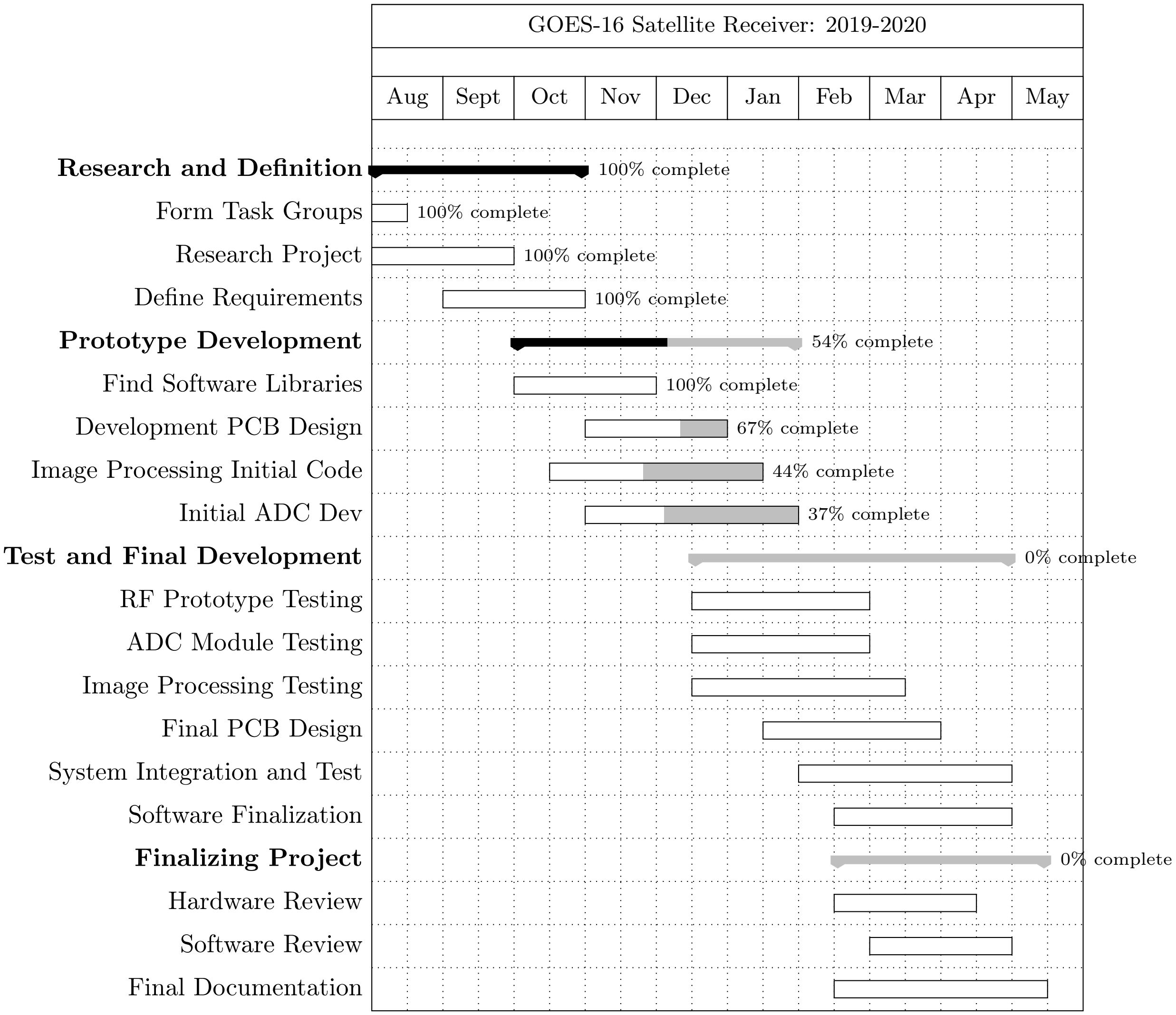

Figure 5: Project Gantt Chart

Along with the Gantt chart shown in Figure 5, the team has defined some dates at which specific deliverables must be completed. Future deliverable dates are subject to change depending on progress. These are as follows:

- RF Team
  - November 19 - Antenna placed in courtyard of Coover Hall
  - December 20 - Development boards designed and ordered
  - February 14 - Working RF hardware prototype including filtering, amplifying, and demodulation
  - March 1 - RF circuit design completed
- ADC Team
  - January 19 - Rev1 of working ADC code
  - February 25 - Tested and working ADC code
  - March 1 - ADC circuit design completed
- Software Team

  - January 13 - Have the code from Open Satellite Project compiled and working properly
  - February 3 - Test the demodulator and decoder with actual data and verify correct operation
  - March 2 - Have RPi4-hosted website operational

- System
  - March 1 - Working prototype with development boards and Rev1 hardware
  - March 10 - PCB designed and ordered for RF and ADC circuits
  - April 1 - Operational system constructed with final prototype
  - April 22 - Reviewed hardware and software documentation released

4.2 Feasibility Assessment

Overall, this project will require an extensive amount of work and knowledge in RF, signal processing, and software systems. As a result, this project will be very difficult for the team to carry out successfully. However, this project is completely feasible given the amount of documentation available online, along with the professional advice of the advisor and other professors at Iowa State University. By separating the team into groups focused on specific sub-systems, team members will become experts in the field they are working are, which will allow constant forward progress in every sub-system of the project.

Initially, the goal of the project was to create a completely weather proof system that can automatically align the antenna, receive the signal, process the signal, and output a high resolution image of Earth. Upon further investigation, a few of the requirements were removed or altered to increase the likelihood of the project being completed successfully. First, it was decided the antenna would be manually aimed to reduce the amount of hardware and software required for the implementation. Next, the team chose to remove the weatherproof requirement, as this requires a significant amount of work to protect hardware from temperature and moisture. With these requirements removed, the focus of the project could be shifted to the RF circuitry, signal processing, and decoding software. Ideally, the team would like a working product complete with a unique PCB design, cables, software, and a mounting system. However, if issues come up while carrying out development, the team may settle for a crude prototype, and include the necessary documentation for producing a final product.

There are a few foreseen challenges that could hinder the development of the final product. One issue that came up initially was the background experience provided by the team. With all electrical engineering majors, there is not a strong background in software, which has proven to require many hours of research. In addition to that, the level of complexity in regards to the signal processing and data decoding is much higher than initially expected. If issues arise when trying to implement the system, it will take much longer to troubleshoot the system due to lack of expertise in the field. That last thing that may hinder progress is lead-times when ordering hardware to prototype. If there are long wait times, development may be slowed down as the team waits for parts to come in.

4.3 Personnel Effort Requirements

Task	Est. Time	Description
Define System Requirements	2 weeks	Initial system and functional requirements must be explored and defined so that the team can begin project research.
Research Domain Topics	4 weeks	Meetings with both the group and Dr. Neihart will direct individual groups into specific research areas. Background knowledge on satellite communications, digital communications, signal processing, etc. must be explored.
RF Component Selection	2 Weeks	Search manufacture offerings to find components that will fill the need of the various RF subsystems.
OpenSat Research	1 weeks	Research the necessary background for implementing the OpenSat functional library.
RF Simulation	2 Weeks	Run simulations of the primary components, (LNA, Mixer, Synthesizer) to determine the passive components required for their proper operation in the systems design.
RF Schematic Creation	2 Weeks	Create the schematics for the different RF boards that are required, ensuring that all of the footprints for the components are correct. Verify all connections with the datasheets for the components.
ADC Selection	2 weeks	Select an ADC platform that covers necessary data and communication requirements. Explore FPGA, DSP, and Microcontroller options.
RF Board Layout	1 Weeks	Layout the RF boards for the initial prototypes. This will require meetings with Dr. Neihart to verify layout choices.
Microcontroller Configuration	2 weeks	Begin configuration of μC. This includes functionality such as SPI, UART, ADC, etc.
RF Board Assembly and Testing	1 Weeks	Assemble the prototype board and perform testing for requirement compliance. This will require the assistance of Dr. Neihart for access to the equipment in the research laboratory.
Compile OpenSat on RPi4	2-3 weeks	Clone source code from the repository and build for use on the Rpi. Ensure that all of the different components of OpenSat are functional and
Microcontroller Testing	2 weeks	Test communication interfaces between IQ demodulator and PLL using an oscilloscope. Verify communication between μC and the RPi4. Ensure ADC functionality using test data.
Configure RPi4-hosted Website	3-4 weeks	Get the website set up and create template for displaying statistics about download.
System Integration	3 weeks	Once all 3 teams are at a good position to do so, we will begin integrating all parts of the project.
System Testing	2 weeks	This will involve recording enough data for an image and seeing if the system can take that data and generate an image.

Table 1: Personnel Effort Requirements

4.4 Other Resource Requirements

Aside from financial resources, the team will be using lab equipment available in Coover such as RF testing equipment, oscilloscopes as well as multimeters to test the functionality of the project.

On top of that, the team will also be using the services of Electronics and Technology Group (ETG) to order the parts needed for the project.

Furthermore, the rooms of Coover are also important for holding weekly team meetings with the team advisor to discuss progress and direction of the project.

4.5 Financial Requirements

This project is provided with a budget of $1,000 USD with the possibility of additional funding if it becomes necessary. The primary financial requirements to the project are to attempt to stay within that budget, and have the final product

5 Testing and Implementation

5.1 Interface Specifications

The team is currently investigating methods of carrying out testing on both the sub-systems and the final product. In order to do this, the team must establish hardware and software connections between the systems and test equipment. The RF team is working on a way to easily test various characteristics of the signal. This requires RF cables and connectors, along with some RF test equipment.

In regards to the ADC and software, the team is developing a way to communicate between the RPi4 and the μC. Once communication between the μC and the RPi4 is established, arbitrary signals will be passed through the μC's ADC to ensure the two components are communicating properly. The ADC team can then interface with software team's RPi4 containing the software for image processing. The software team can use sample image data, not necessarily GOES satellite data, and confirm that the software is functioning properly and producing an image as expected. Once we have all sub-systems operating properly, we can move forward and interface all systems together.

Sample image data can be found at: https://www.micasense.com/altum-sample-data

5.2 Hardware and Software

Testing each sub-system and the final product will require the use of a few hardware and software tools. These tools are listed and outlined below:

Raspian OS - The OS that the team is using for the RPi4. It includes a terminal where the team will be running the code on. The terminal is also used to install the necessary personal package archives needed to compile the code.
Oscilloscope - The oscilloscope is an instrument used to measure and analyze waveforms of signals. The oscilloscope was used to verify the functionality of the μC's SPI through the MOSI, MISO, CLK and SS pins.
SDR - The SDR will be used as a way to test the signal reception of the antenna. It will act as a simple spectrum analyzer and the team will use it to confirm the antenna is aimed in the correct position, and that the desired signal is being received with the antenna.
Spectrum Analyzer - A spectrum analyzer will be used to test properties of the RF circuit. It will be used as a way to estimate the total attenuation in the circuit, along with confirming that all filtering and down-mixing is successful.
Serial Monitor - The team will use a serial communications monitor software, such as PuTTy, to ensure

signal processing taking place in the ADC is working correctly. The serial monitor will allow the team to troubleshoot and test the code in the μC during run-time.

5.3 Functional Testing

Functional testing up until this stage of the project is limited to testing sub-systems for proper operation. Each of the sub-systems are currently being tested individually. The RF team is performing tests on the antenna to make sure the desired signal is being captured. The ADC team is currently testing code with the μC module to confirm no errors when uploading and running code. The software team has been testing the functionality of the teams desired code library on the RPi4.

The team will conduct further testing in the next semester of the project. Follow-on testing will occur once the team is fully comfortable with integrating team efforts. This document will be updated as tests are being conducted.

5.4 Non-Functional Testing

Since the end product is images of the earth, we can easily check how well our system performs by looking at the image quality. However, it will be difficult to know where the errors are happening by just looking at the images. The software from the Open Satellite Project has GUIs built that show performance data such as bit error counts that will help determine where the errors are happening.

5.5 Process

The team is still developing the initial prototype and has not begun detailed testing of the system. This section of the report will be updated as functional and non-functional testing occurs.

5.6 Results

This section will be updated once the testing plan has been carried out in the second semester. Detailed results will be posted once the product has been thoroughly tested.

6 Closing Material

6.1 Conclusion

At the conclusion of the first semester of Senior Design, the team is in a good position to be able to finish the project on time. The RF team has designed much of the hardware and is working on a PCB board layout to implement the design. The ADC team has developed a clear goal and is working on getting the μC and RPi4 to communicate. The software team has gotten the code for the demodulator and decoder on the RPi4 and is working on getting the project to compile.

The exact implementation of the system that is necessary to download and reconstruct the images of earth that we are using is slightly different than what others have done before, but the ideas are all the same because the signal is encoded and transmitted in a standard format, so the same processing will need to be done in any implementation. Our system should work just as well as any other system to do this job.

6.2 References

Works Cited

Brettbolt, Luigi Freitas Cruz; Lucas Teske; “Open Satellite Project”. *GitHub repository* (2019). https://github.com/opensatelliteproject.

Razavi, B. *RF Microelectronics*. Prentice Hall, 2012. Web. Prentice Hall communications engineering and emerging technologies series.

Roberson, Jeremy, et al. “A Flexible Software Based EMWIN/HRIT Prototype Solution for the GOES-R Transition”. *ATR-2010 Aerospace report* 2 (2009). Print.

Sklar, B. *Digital communications: fundamentals and applications*. Prentice-Hall PTR, 2001. Web. Prentice Hall Communications Engineering and Emerging Technologies Series.

6.3 Appendices

Nothing to add here as of yet.