Distributed Ground Station Network For CubeSat Communications

Zachary J. Leffke

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Robert W. McGwier, Chair
Dennis G. Sweeney
Scott S. Bailey

December 6, 2013
Blacksburg, Virginia

Keywords: Cubesat, Software Defined Radio, Ground Station, Distributed Network
Copyright 2013, Zachary J. Leffke

Distributed Ground Station Network For CubeSat Communications

Zachary J. Leffke

(ABSTRACT)

In the last decade the world has seen a steadily increasing number of Cube Satellites deployed to Low Earth Orbit. Traditionally, these cubesats rely on Amateur Radio communications technology that are proven to work from space. However, as data volumes increase, the existing Amateur Radio protocols, combined with the restrictions of use for the Amateur Radio Spectrum, as well as the trend to build one control station per cubesat, result in a bottle neck effect whereby existing communications methods are no longer sufficient to support the increasing data volumes of the spacecraft.

This Masters Thesis work explores the concept of deploying a network of distributed ground station receiver nodes for the purposes of increasing access time to the spacecraft, and thereby increasing the potential amount of data that can be transferred from orbit to the ground. The current trends in cubesat communications will be analyzed and an argument will be made in favor of transitioning to more modern digital communications approaches for on orbit missions. Finally, a candidate ground station receiver node design is presented a possible design that could be used to deploy such a network.

Dedication

This work is dedicated to my loving wife Sarah. Without her endless support, this work would not have been possible.

It is also dedicated to our collection of animals (aka children, listed in order of 2 dogs, 6 cats, 2 hairless guinea pigs, and 2 fish): Katie, Bing, Peaches, Puff 'n Stuff, Paul, Peepers, Big Nose, Shrub, Poppy, Blossom, Merle, and Gus.

Acknowledgments

I would like to thank Dr. Robert W. McGwier for his steady guidance in this work. His depth of experience in Amateur Radio satellite communications and Software Defined Radio has been an invaluable resource.

I would also like to thank the Radio Amateur Satellite Corporation, commonly known as AMSAT. This collection of eclectic geniuses, that build and launch satellites as part of a hobby, have been a steady source of encouragement and technical savvy and are an excellent group of role models that I am proud to be associated with.

Specifically, I would like to thank Douglass Quagliana and Joe Fitzgerald from AMSAT for their specific words of encouragement and genuine enthusiasm for this project. I hope the work presented here can help further the mission of AMSAT.

I would also like to thank Howard Long, the inventor of the FunCube Dongle Pro Plus. Without this economic and brilliantly designed device, the creation of the prototype presented in this work would not have been possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the increased access to Space for Cube Satellite (cubesat) developers through the NASA Educational Launch of Nano-Satellites (ELaNa) initiative in the United States, as well as other international launch opportunites, the world has seen a steadily increasing number of cubesats deployed to Low Earth Orbit in the last decade. These spacecraft are being built by University, Government, and Commercial institutions for the purposes of performing Space-based research and rely heavily on Amateur Radio communications technology. Though it is true that both tend to occupy Amateur Radio spectrum, an important distinction must be made between cubesats designed for performing space research and amateur radio satellites. For the purposes of this document the term “cubesats” will refer to spacecraft with the mission objective of collecting data while on orbit for the purposes of supporting Space-based research. On the other hand, the term “Amateur Radio spacecraft” will refer to spacecraft with the mission objective of providing a communications relay service to licensed Amateur Radio operators around the world.

The most common mission objective of Amateur Radio spacecraft is to provide a voice and/or data relay service between two Amateur Radio operators on the ground. This is primarily accomplished by launching a communications payload on a Low Earth Orbit spacecraft. The communications payload usually consists of an uplink receiver designed to operate in one frequency band connected to a downlink transmitter designed to operate in a different frequency band. This method of operation is very similar to what is known as a cross band repeater. The frequencies utilized by most Amateur Radio spacecraft reside in what is known as the Amateur Satellite Service, a subset of the Amateur Radio frequency allocations.

Cubesats, though sharing similarities with Amateur Radio spacecraft, have very different mission objectives. Instead of carrying a communications relay payload, these spacecraft carry scientific payloads. These scientific payloads generally consist of some type of scientific instrumentation or sensor(s). Data is collected from the payload and stored onboard the spacecraft until such a time, as allowed by the orbit of the spacecraft, that the data can be downloaded to the operating control station.

This document will address three specific areas. First the communications protocols currently in use for cubesats are antiquated and need to be replaced by more modern digital communications approaches. Second, the concept of deploying a receive only ground station network for the purposes of increased access time will be explored. Finally, a candidate ground station design is presented as a possible enabler for the deployment of the ground station network. If these three areas are combined, more efficient modulation schemes (with forward error correction), the deployment of a network of ground station receiver nodes, and the candidate design (or something similar) is utilized for the ground station node, cubesats will be able to transfer larger amounts of data to the ground, all while continuing to use low rate, and thus low bandwidth, communications protocols.

Chapter 2

Background

On December 12, 1961 OSCAR-1 was successfully deployed as a secondary payload from Discoverer 36 into Low Earth Orbit and was received by over 570 Amateur Radio operators around the world during its 22 day lifetime [1]. The designation OSCAR is still in use today and stands for "Orbiting Spacecraft Carrying Amateur Radio." This marked the beginning of over 60 years of space operations for Amateur Radio enthusiasts. As Universities and other institutions began to develop cubesats and smallsats for the purposes of Space based research it is no surprise that they turned to the Amateur Radio community for technological guidance.

Cubesats, like all spacecraft, rely heavily on wireless communications technology. As with most trends in wireless communication, there exists a demand for increased data rates. As new cubesats are being developed and deployed there is a noticeable trend of increasing data volumes generated by the spacecraft. This increase in data volumes is primarily attributed to more sensitive scientific instruments with higher resolutions, increased duty cycle in the use of the instruments, and combining more and more sensors into a single payload.

Generally speaking the communications method used is not always a primary concern in the design of the spacecraft, and there is a tendency to simply duplicate methods utilized by previous missions. This is not altogether a bad mentality as the use of components with "flight history" is usually a good idea. Principal Investigators simply assume that they can use existing communications technology to get the data to the ground, and if they select a radio with flight history, they reduce the probability of mission failure. This is generally a true statement, however it introduces a problem.

To date the majority of spacecraft flown utilize Amateur Radio frequency allocations and protocols common to Amateur Radio operations [2]. The problem that exists is that Amateur Radio communications are limited by FCC regulation as to the amount of spectrum a single user (or spacecraft) is allowed to occupy. Therefore, as data volumes for these space science cubesats increase, previously acceptable communications protocols that are bandwidth limited (by regulation) now become the bottleneck in getting the mission science data to the ground. This is compounded by the problem of the trend to build a single ground station for each spacecraft. The deployment of a number of networked ground station receiver nodes would increase access time to the spacecraft and ultimately allow for increased amounts of science data to be transferred to the ground.

2.1 What is a CubeSat?

Cubesats belong to a class of satellites known as pico-satellites, a subset of what is referred to as smallsats. Cubesats are extremely small satellites. Dimensions are referred to in “cubesat units” called a U. A 1U cubesat is a cube that measures 10 cm on an edge and weighs less than 1.33 kg. The most common cubesat sizes range from 1U to 3U, where 3U is 10x10x30 cm and weighs less than 4.0 kg. Recently, 6U designs have come onto the scene, and measure 10x20x30 cm and are referred to as “six-pack” designs. The cubesat standard was developed by researchers at California Polytechnic State University (Cal Poly) and Stanford University.

Cubesats are delivered to orbit as secondary payloads that provide ballast during the launch of a primary satellite. Once the primary satellite has been safely separated from the launch vehicle, the secondary payloads are delivered to orbit. The most common deployment mechanism for a cubesat is the Poly Pico Orbital Deployer (P-POD). A P-POD can hold up to 3 Us worth of cubesat (3 1U cubesats, or 1 3U cubesat). A single launch vehicle is capable of holding multiple PPODs, allowing a large number of cubesats to be delivered to orbit during a single launch.

Because of the small form factor of cubesats, they are severely limited in their capabilities. The small surface area of any given cubesat limits the amount of power that can be generated by the Solar Cells. This single constraint drives many of the design decisions for the spacecraft. A delicate balance must be achieved by the mission designers and power has to be carefully utilized. In spite of this severe limitation, cubesats have incredibly sophisticated subsystems and payloads are becoming more complex with each subsequent launch. Some of the subsystems common to all cubesats include communications, power, stabilization, and of course the payload. Power must be distributed carefully to these subsystems to ensure mission success. It is the communications subsystem that this work will analyze in detail.

A general rule in communications theory concerning Antenna design is that the smaller the antenna, the more omni-directional the radiation pattern and the lower the gain. Cubesats tend to have very simple antenna systems. At the simplest level, monopoles are used. A slightly more complex design that is frequently used is called a canted turnstile. This design includes the use of a 4 quarter wave elements. When deployed the antenna takes on the geometry of two V-dipoles oriented 90 degrees from each other. When fed 90 degrees out of phase these two dipoles generate a circularly polarized radiation pattern. The canted, or V, nature of the antennas help to direct radiation of the antenna, resulting in a slight gain increase.

2.2 Current Cubesat Trends

This section will examine current trends in cubesat communications. Appendix A contains Table A.1, a list of currently active cubesats that occupy Amateur Radio spectrum [3]. It is this table that will be used to highlight trends in cubesat communications, specifically in selection of frequency and modulation scheme. It is true that there are operational cubesats that exist outside of Amateur Radio frequencies, but those spacecraft are not the focus of this Thesis work and the focus will be on spacecraft that reside in Amateur Radio frequency allocations.

2.2.1 Frequency Allocations

The most common bands for the Amateur Satellite Service are the two meter (2m a.k.a. VHF) and seventy centimeter (70cm a.k.a. UHF) bands [2]. Though the Amateur Radio allocation for the 2m band is 144.0 - 148.0 MHz, the Amateur Satellite Service sub-band allocation is 144.0 to 146.0 MHz, with a preferred operating sub band of 145.8 - 146.0 MHz. Similarly, though the 70cm Amateur Radio Allocation is 420.0 - 450.0 MHz, the Amateur Satellite Service sub-band allocation is 435.0 - 438.0 MHz. Licensing of these spacecraft is accomplished through the Federal Communication Commission and the spacecraft radio operations are governed by Title 47, Part 97 of the Code of Federal Regulations [4]. Figure 2.1 below shows the percentage of use of these frequencies.

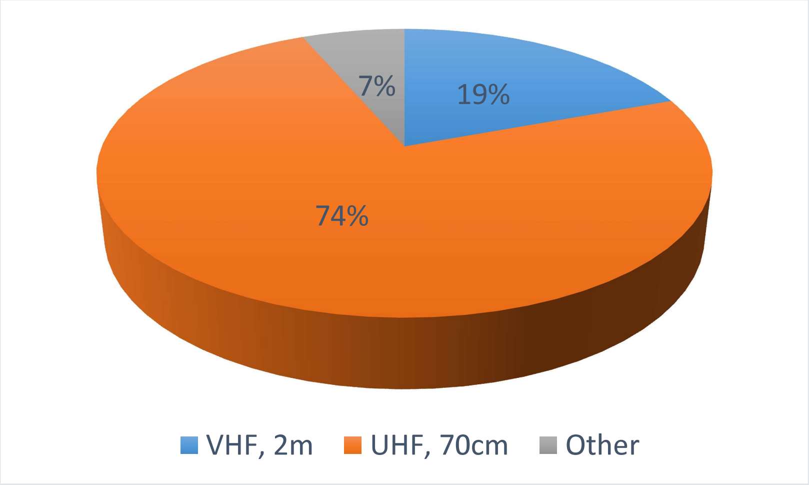


Figure 2.1: Frequency Use.

2.2.2 Modulations

Also shown in Table A.1 are the modulation schemes utilized for the downlink and satellite beacons. Two of the most common modulation schemes employed by the currently active cubesats are Audio Frequency Shift Keying (AFSK) and Frequency Shift Keying / Gaussian Minimum Shift Keying (FSK/GMSK). Figure 2.2 below shows the percentages of use. The "Other" modulation schemes depicted in the figure include a mixture of less frequently used modulation schemes mostly consisting of Single Side Band (SSB) Amplitude Modulation (AM) and Frequency Modulation (FM) used for voice operations by the satellites that provide a voice relay service to licensed Amateur Radio operators. Also included in the "Other" grouping is the CW (a.k.a. Morse Code) beacons utilized by some of the spacecraft. A more detailed analysis of the modulation schemes employed by existing cubesats is explored in Chapter 3.

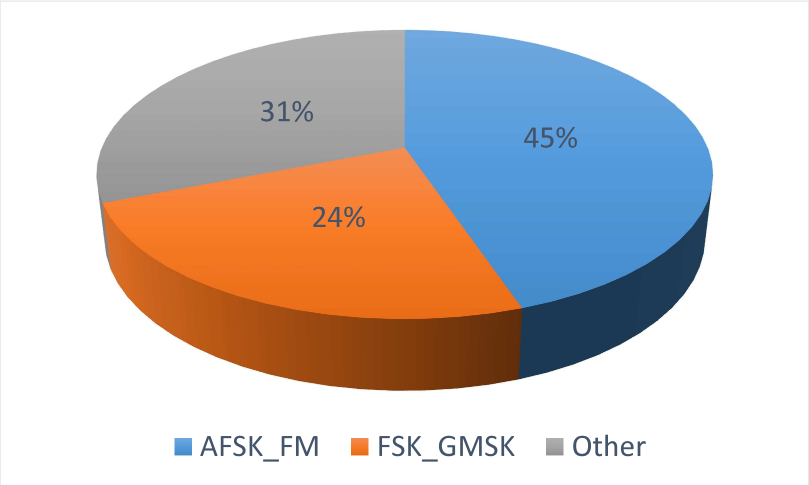


Figure 2.2: Commonly Used Modulation Schemes.

2.3 FCC Regulation

The Federal Communications Commission (FCC) is responsible for the radio licensing of all non-Federal spacecraft. The National Telecommunications & Information Administration (NTIA) is responsible for all radio licensing of Federal spacecraft. Recently, the FCC released a notification concerning guidance for licensing of cubesats and smallsats [5]. This guidance answers many questions for cubesat developers including what legislation governs the radio subsystems of the spacecraft. There are three procedures for licensing of a spacecraft under FCC (non-Federal) rules as set forth in Title 47 of the Code of Federal Regulation [5]. The first, and most common, procedure is Title 47, Part 25, Satellite Communications. This section governs licensing for most commercial satellites, for example the geostationary satellites that provide satellite TV service (e.g. EchoStar) or communications services (e.g. IntelSat). Second, if occupying Amateur Radio spectrum for the purposes of providing a service to the Amateur Radio community then the spacecraft is governed by Title 47, Part 97, Amateur Radio Service. If, however, the spacecraft occupies Amateur Radio spectrum for the purposes of performing research, then an Experimental License is obtained and operation is governed by Title 47, Part 5, Experimental Radio Service.

Therefore, to date, most of the more recent cubesats that perform Space-based research are being licensed under the CFR, Title 47, Part 5. However, these spacecraft are using radio subsystems that utilize technology derived from spacecraft built by Amateur Radio operators for the purposes of providing voice and data relay services to the Amateur Radio community, and thus use the same frequencies and modulation schemes. Therefore, though they have different classes of FCC licenses, the technology used in both spacecraft radio subsystems generally adhere to the guidelines and limitations set forth by the CFR, Title 47, Part 97 Rules (which is more restrictive). The choice of an Experimental License over an Amateur Radio License is more of an administrative way of keeping track of the spacecraft radio systems by the FCC than an actual engineering decision. In practice, the Principal Investigators do not care how they get licensed as long as they can transmit their payload data to the ground without legal hassles. It is these CFR, Title 47, Part 97 emission limitations that influence the current cubesat communications technology and restrict the amount of spectrum that can be occupied by a single user (or spacecraft).

As stated in sections 2.2.1, the most used frequency bands are the Amateur UHF (70cm) band and the Amateur VHF (2m) band. As stated in Section 2.2.2 the most frequently used modulation scheme is AFSK followed by FSK/GMSK. These two modulation schemes for the specific bands, do not violate the limits set forth in the CFR, Title 47, Part 97 Rules. Specifically, the CFR, Title 47, Part 97.305 Authorized Emission Types, Subsection c, defines the frequencies available for use and the types of authorized emissions allowed per frequency band. The emission types are defined in the CFR, Title 47, Part 97.307 Emission Standards, subsection f.

Per CFR, Title 47, Parts 97.305(c) and 97.307(f) the authorized bandwidth for data communications in the VHF (2m) band is up to 20 kHz, with a symbol rate of 19.6 ksps or less [4]. Per CFR, Title 47, Parts 97.305(c) and 97.307(f) the authorized bandwidth for data communications in the UHF (70cm) band is up to 100 kHz, with a symbol rate of 56.0 ksps or less [4]. Although a bandwidth of 100 kHz is authorized in the UHF band, it is important to remember that spacecraft licensed under this section of the Code of Federal Regulations are generally intended to provide a voice or data relay service to the Amateur Radio Community. As described in Chapter 1, these spacecraft tend to operate like cross-band repeater systems, with a simple frequency translation between the uplink and downlink frequencies (usually using 2m and 70cm frequencies). Therefore, the bandwidth occupied by any single user transmission must adhere to the most limiting restriction, 20 kHz. Since cubesat data radios rely heavily on technology from the Amateur Radio Service, even though most of them operate in the UHF (70cm) band and are licensed under Part 5 rules, the bandwidth of the radios are generally limited to 20 kHz of spectrum.

Chapter 3

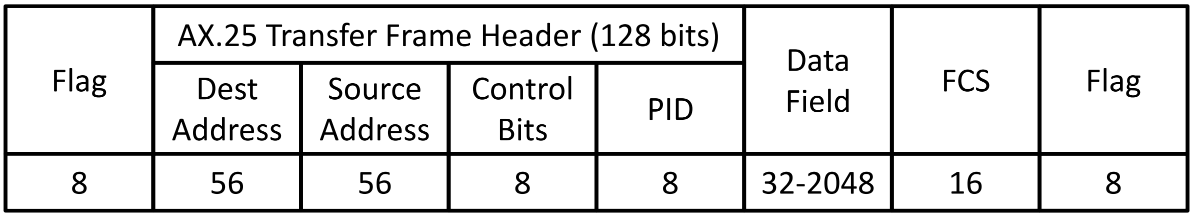
Communication Protocol Analysis

This Masters Thesis work calls for a change in the modulation schemes utilized for cubesat communications. Additionally, a network of ground station receiver nodes is presented in Chapter 4 and number of simulations are conducted. In order to thoroughly understand the amount of data that can be transferred from orbit to the ground by this notional network, it is important to understand the link layer protocol and the amount of overhead associated with it. It is also important to thoroughly understand the modulation schemes utilized by cube satellites. The most common data link layer protocol utilized by existing cubesats is the AX.25 protocol. This is examined in more depth in Section 3.1 below. This is then followed by an analysis of the current modulation schemes in use by the cubesats in Section 3.2. Finally, the impact of switching from the current analog modulation schemes to digital modulation schemes is examined in Section 3.3.

3.1 AX.25 Protocol

The most common link layer protocol utilized by existing cubesats is the AX.25 version 2.2 Link Layer Protocol. This protocol is frequently used by Amateur Radio Packet transmissions. This protocol conforms to the International Standards Organization Information Standards 3309, 4335, and 7809 High-level Data Link Control (HDLC) protocol [6]. It is also influenced by the Consultative Committee in International Telegraph and Telephone (CCITT) Recommendations Q.920 and Q.921 (LAP-D) protocol [6].

Specifically, the Unnumbered Information Frame (UI-frame) format of the AX.25 protocol is utilized in most cubesat communications schemes. It is important to determine the potential overhead of this frame format in order to accurately determine the amount of actual data that can be transferred from the payload on orbit to the ground. Figure 3.1 below shows the structure of the AX.25 UI-frame, as described in [7].



Flag	AX.25 Transfer Frame Header (128 bits)				Data Field	FCS	Flag
	Dest Address	Source Address	Control Bits	PID			
8	56	56	8	8	32-2048	16	8

Figure 3.1: AX.25 Unnumbered Information Frame (UI-Frame) Structure.

As shown by the bit counts in the figure above, the maximum size of the AX.25 UI-frame is 2208 bits. The maximum length of the data field, which is where actual payload data would be contained, is 2048 bits. Application of equations Eq. 3.1 and Eq. 3.2 below, substituting 2048 for the data bits and 2208 for the total bits, yields an efficiency of 92.75% and thus an overhead of 7.25% for the AX.25 UI-frame data link layer protocol. It is important to note that this is the *maximum* efficiency and thus *minimum* overhead. This assumes the full length of data field is utilized for each transmission. This assumption will carry through the rest of this document, including the simulations presented in Section 4.2.

$$Efficiency, \eta = \frac{data\,bits}{total\,bits} \tag{3.1}$$

$$Overhead = 1 - \eta \tag{3.2}$$

A number of companies have developed and sold devices known as Terminal Node Controllers (TNCs) that are compliant with the AX.25 link layer protocol. These devices are very similar to dialup modems. Essentially, TNCs are capable of modulating/demodulating the signals transmitted/received by existing Amateur Radio equipment, specifically the common FM radio. For the lower rate AFSK communications, the audio connector of the radio is connected to the TNC. For higher rate FSK/GMSK communications the FM radio's varactor input on the FM modulator (for transmit) and the discriminator output (for receive) are connected to the TNC. Both AFSK and FSK/GMSK modulation methods, which are very common in cubesat communications, are described in more detail in Section 3.2 below.

3.2 Modulation Analysis

This section examines the modulation schemes currently in use by active cubesats. Figure 3.2 is perhaps the most revealing figure in this entire document and shows the required Carrier to Noise Ratio (CNR) for a set of modulation schemes. As a common reference, all modulation schemes are analyzed with a 10^{-5} target Bit Error Rate (BER). The required CNR for both AFSK_FM and G3RUH_FSK were obtained from a Link Budget Calculator developed by AMSAT in conjunction with the International Amateur Radio Union [8]. The rest of the modulation schemes are the well known theoretical error curves for a given modulation scheme.

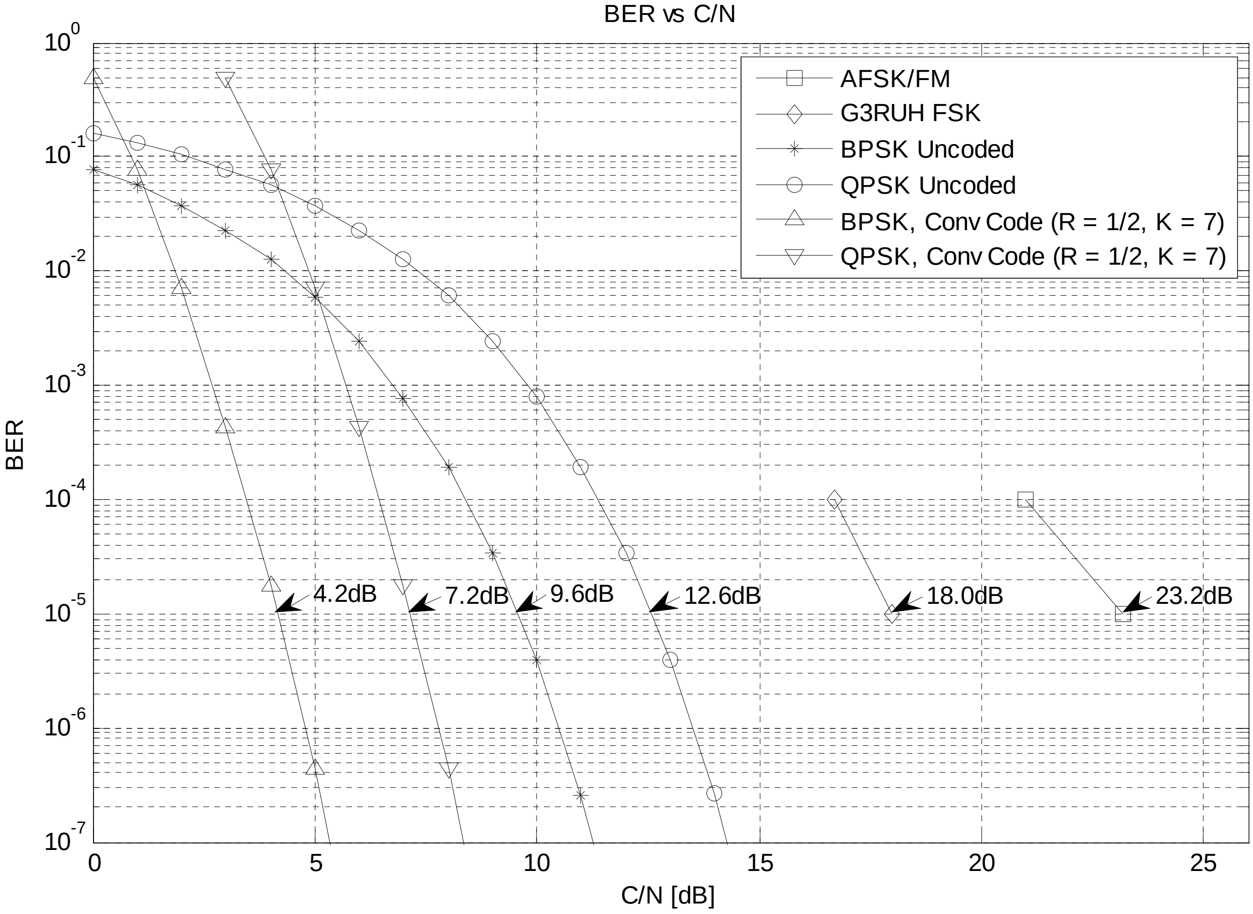


Figure 3.2: Bit Error Rate (BER) vs. Carrier to Noise Ratio (CNR).

3.2.1 Audio Frequency Shift Keying, AFSK_FM

Audio Frequency Shift Keying is a form of modulation that uses the baseband signal (the bits from the AX.25 UI-Frame) to differentially alternate audio tones at 1200 Hz and 2200 Hz. These audio tones are then passed to an analog FM Radio's audio input to modulate the FM carrier. This modulation scheme will hence be referred to as AFSK_FM. The most common data rate associated with this modulation scheme is 1200 bits per second (bps) and is also referred to as AFSK1200. This method of modulation is limited by the audio filtering in many of the commonly available radios. Since the audio is filtered between the microphone and the FM modulator, the amount of carrier deviation, and thus the data rate of the radio is limited to the practical rate of 1200 bps. As stated in Section 2.2.2, AFSK_FM is the most common modulation scheme employed by cubesats and Figure 3.2 shows that for a Bit Error Rate of 10^{-5}, a Carrier to Noise Ratio of 23.2 dB is required. Thus the most common modulation scheme employed by cubesats requires the highest CNR.

3.2.2 G3RUH Frequency Shift Keying, G3RUH_FSK

Frequency Shift Keying is a form of modulation that directly modulates the radio's center frequency as a function of the input baseband signal. Specifically, in Amateur Radio satellite operations, a form of FSK is employed known as G3RUH FSK. This modulation scheme, developed by James Miller, Amateur Radio callsign G3RUH, is a form of FSK that utilizes currently available FM Radios to modulate/demodulate FSK signals [9][10]. As described above, G3RUH FSK modulation is accomplished by directly connecting the TNC to the FM modulator/demodulator.

This method of modulation effectively bypasses the audio filtering circuitry of the existing radios. In conjunction with radios that have suitably wide enough IF bandwidths, this allows higher data rates to be achieved over AFSK methods. The most common data rate associated with this modulation scheme is 9600 bps. It should also be noted that Gaussian Minimum Shift Keying, a form of FSK, is also used in cubesat communications. This modulation scheme will hence be referred to as G3RUH_FSK. As stated in Section 2.2.2, G3RUH_FSK is the second most common modulation scheme employed by cubesats and Figure 3.2 shows that for a Bit Error Rate of 10^{-5}, a Carrier to Noise Ratio of 18.0 dB is required. Thus the second most common modulation scheme employed by cubesats requires the second highest CNR.

3.2.3 Digital Modulation Schemes

The modulation of the phase of a signal is known as Phase Shift Keying (PSK). Specifically, Binary Phase Shift Keying (BPSK) and Quadrature Phase Shift Keying (QPSK) are examined as alternatives to the current analog modulation schemes. An important note must be made concerning BPSK and QPSK. Most textbooks describe the required Energy per Bit over Noise (Eb/No) for both BPSK and QPSK for a target Bit Error Rate (BER) as identical. This is because QPSK can be viewed as two BPSK signals transmitted in the same channel, but in phase quadrature, and thus not interfering with each other. This is in general true, but when analyzing satellite communication links the Carrier to Noise Ratio (CNR), not the Energy per bit over Noise (Eb/No) is utilized for link budget analysis [14]. For modulation schemes with one bit per symbol (e.g. BPSK), the Eb/No is equivalent to the CNR. Since QPSK is two bits per symbol, there is a factor of two difference from BPSK, thus QPSK requires twice the amount of energy to achieve the same bit error rate. The specific equations for Probability of Error for both BPSK and QPSK as a function of Carrier to Noise Ratio (not Eb/No), as described in [14], are shown in Equations 3.3 and 3.4 below (respectively) and are plotted in Figure 3.2.

$$P_{e,BPSK} = \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{C}{N}}\right) \tag{3.3}$$

$$P_{e,QPSK} = \frac{1}{2}\,\mathrm{erfc}\left(\sqrt{\frac{C}{2N}}\right) \tag{3.4}$$

3.3 Switching from Analog to Digital

While both AFSK_FM and G3RUH_FSK modulation techniques have the benefit of working with a plethora of existing Amateur Radio FM equipment currently in use by the community, they do have drawbacks. The G3RUH_FSK modulation method is not the same as traditional Non-Coherent or Coherent FSK. G3RUH_FSK requires higher Carrier to Noise Ratios than traditional FSK (both coherent and non-coherent) in order to achieve uncorrupted data transfer. Because both rely on the use of analog FM radios to accomplish the modulation/demodulation, they are both still technically an analog form of communication, even though digital data is being transferred. It is this use of analog communications techniques, that require high CNRs, that should be replaced by digital modulation techniques. Specifically, this Thesis work examines the use of BPSK and QPSK as alternatives to AFSK_FM and G3RUH_FSK.

3.3.1 Occupied Bandwidth and Achievable Data Rates

As described in Section 2.3 the FCC requires the occupied bandwidth to be less than 20 kHz for the 2m band and less than 100 kHz for the 70cm band. In practice currently active cubesats restrict their operations to occupying 20 kHz or less. Given these limitations, further analysis will assume a fixed occupied bandwidth of 20 kHz.

For BPSK and QPSK the relationship between Occupied RF Bandwidth and Symbol Rate is defined by Equation 3.5 from [14], where α represents the excess bandwidth factor, and R_s is the Symbol Rate. In BPSK and QPSK modulators, Root Raised Cosine Filters are employed as pulse shaping filters to limit the actual bandwidth of the signal. The excess bandwidth parameter represents the filter roll off of these pulse shaping filters. Practically speaking, α can have a range of values depending on a number of factors. For this Thesis work, an arbitrarily assumed value of 0.35 will be used for all calculations and simulations. Equation 3.6 shows the relationship between Symbol Rate (R_s) and Bit Rate (R_b, a.k.a. Data Rate), where n is the number of bits per symbol.

$$BW_{occupied} = R_s * (1 + \alpha) \tag{3.5}$$

$$R_b = R_s * n \tag{3.6}$$

Fixed Bandwidth, Variable Data Rate

Given the fixed RF Bandwidth of 20 kHz, an assumed α of 0.35, and rearranging Equation 3.5 a maximum symbol rate of approximately 14,815 symbols per second can be achieved. This falls below the 19.6 ksps limit set forth by the FCC and describe in Section 2.3. For uncoded BPSK, which has one bit per symbol, this results in a maximum data rate of 14.815 kbps. For uncoded QPSK, which has two bits per symbol, this results in a maximum data rate of 29.630 kbps. Both of these data rates are *higher* than G3RUH_FSK at 9.6 kbps while simultaneously requiring *lower* Carrier to Noise Ratios. Stated differently to highlight the importance of this, for a fixed RF bandwidth of 20 kHz, one G3RUH_FSK signal at 9.6 kbps can be transmitted, or one QPSK signal can be transmitted at 29.630 kbps, over three times faster than G3RUH_FSK.

Fixed Data Rate, Variable Bandwidth

Another way to look at this problem is to consider the required downlink rate and then determine how much bandwidth must be occupied to achieve it. For 9.6 kbps using G3RUH_FSK about 20 kHz of RF spectrum is required. For BPSK, with one bit per symbol, the required symbol rate to achieve 9.6 kbps is 9.6 ksps. Again, with an α of 0.35, this results in an occupied RF spectrum of 12.96 kHz. For QPSK, at 2 bits per symbol, the required symbol rate to achieve 9.6 kbps is 4.8 ksps. Again, with an α of 0.35, this results in an occupied RF bandwidth of 6.48 kHz. When viewed in this context, for a fixed data rate of 9.6 kbps, and an assumed α of 0.35, either one G3RUH_FSK signal can reside in the given 20 kHz of RF bandwidth or three separate QPSK signals could reside in the same 20 kHz of RF bandwidth. If the required data rate were 1.2 kbps, then for QPSK, the required symbol rate would be 600 sps and the occupied RF bandwidth would be 810 Hz. For a 20 kHz RF channel, fully 24 signals operating at 1.2 kbps using uncoded QPSK could occupy the same RF Spectrum.

3.3.2 Forward Error Correction

Figure 3.2 above shows two more error curves other than AFSK_FM, G3RUH_FSK, BPSK, and QPSK. These two error curves are for Convolutionally Coded BPSK and QPSK with Rate=$\frac{1}{2}$, a depth of K=7, and using soft decision Viterbi decoders. The resulting Carrier to Noise Ratio, for a target BER of 10^{-5}, for BPSK is 4.2 dB and for QPSK is 7.2 dB.

This method of forward error correction results in a decrease in data rate by a factor of two. Because this method of modulation is using R = $\frac{1}{2}$, for every data bit, two raw bits are transmitted. This means that for a fixed raw data rate, say 9.6 kbps, the actual amount of payload data transferred would be 4.8 kbps. Again this is because every two bits transmitted represent 1 actual data bit. In order to achieve an actual payload data rate of 9.6 kbps, the radio would have to transmit at a raw data rate of 19.2 kbps.

3.3.3 Impact on the Link Budget

Simply by switching modulation schemes to BPSK or QPSK, which requires 9.6 or 12.6 dB of CNR (respectively), significant gains can be achieved. Specifically, switching from AFSK_FM to BPSK results in 13.2 dB of gain and switching from G3RUH_FSK to BPSK results in a gain of 8.4 dB. Additionally, the use of Forward Error Correction will further reduce the required signal to noise ratio and thus increase the gains over use of AFSK_FM and G3RUH_FSK. To show one such option, Figure 3.2 depicts the error curve for BPSK using a Convolutional Code with a depth of K = 7 and Rate 1/2 using soft decision decoding. The required CNR for this scheme is 4.2 dB, representing a further gain of 5.4 dB over uncoded BPSK. This equates to a gain of 19.0 dB over AFSK_FM and a gain of 13.8 dB over G3RUH_FSK.

Interestingly, the antennas commonly used to communicate with Amateur Radio satellites, such as the Arrow Antenna, have about 10 - 12 dBi of gain as shown in Figure 5.13 in Section 5.2.3. If these satellites were to use BPSK or QPSK modulations then the need for directional antennas, and thus tracking, would be eliminated. Instead, antennas with hemispherical radiation patterns, and around 2 to 3 dBi of gain could be utilized effectively. Antenna designs include the Quadrifilar Helix, crossed Moxons, the Crossed Loop antenna (a.k.a the Eggbeater), and a number of other candidate designs.

Chapter 4

Network Design

Ground Station Networks are not a new concept. The Amateur Radio community itself has acted as a distributed network for decades. Research into the implementation of existing ground station networks has been performed by multiple institutions [11][12]. The Global Educational Network for Space Operations (GENSO) was an attempt to create a standardized ground station network utilizing the existing Amateur Radio community [13]. This section will present a ground station network design for 19 different locations around the world, consisting of functionally identical receive only nodes. Additionally, four spacecraft have been selected from the active satellites list in Appendix A that are representative of the larger set of the types of orbits that can be expected for a Low Earth Orbit cubesats. A simulation is then conducted using AGI's Systems Tool Kit (formerly Satellite Tool Kit, a.k.a. STK). The simulation is conducted for a one month period between January 1, 2014 0000 UTC to February 1, 2014 0000 UTC.

4.1 Network Requirements

4.1.1 Cubesat Factors

Cubesats are severely limited in Size, Weight, and Power (SWaP). These limitations drive design requirements for the spacecraft that must be taken into account for the ground station network. The challenge in designing a network that can interact with multiple cubesats is that there are a large number of possible solutions when designing a cubesat. That being said, there are certain commonalities between cubesats that must be factored into the ground station network design.

First, cubesats tend to have relatively simple antenna designs. Simple monopoles and dipoles are frequently used. Patch antennas are also sometimes utilized for the higher frequencies of operation, such as S-band. The canted turnstile design is a combination of two V dipoles that generates a circular radiation pattern. All of these antennas have basically omni-directional, or near omni-directional radiation patterns. This limitation is important as it defines the RF footprint of the spacecraft. Essentially, because of the omni-directional nature of the antenna system, combined with the limited stabilization capability of the spacecraft, the visible footprint of the spacecraft is equivalent to the RF footprint of the spacecraft. When viewed from orbit, there is a defined area on the earth that has line of site with the spacecraft, known as the satellite's footprint. If the satellite has line of site with a point on the earth, it will radiate RF energy to that point. This effect results in significant overlap for receiving ground locations.

Cubesats are small, and thus have limited solar cell area and power generation capabilities from their solar cells. A portion of this energy is consumed by the radios in the communications subsystems of the cubesats. Because of the limited power, cubesats tend to radiate low amounts of power. On one extreme, some Amateur Radio spacecraft transponders are capable of generating over 1 Watt of power input to the antenna terminals. However, for research cubesats, the primary payload is generally *not* the communications transponder and is instead some type of sensor system designed to collect data. Generally, power is redistributed from the radio to the payload. This reduces the amount of power the cubesat has available to deliver to the radio. Generally speaking, cubesats with research oriented missions generate a few hundred milliwatts or less of RF energy input to the antenna terminals.

4.1.2 Spatial Diversity

Because of the omni-directional nature of the antenna systems onboard cubesats they tend to have very broad footprints. This means that for any two ground station locations inside the footprint of a cubesat, when a packet is transmitted, both ground locations receive the same information (assuming both locations are able to properly demodulate the received signal). Therefore, when analyzing the capacity of the network, overlap between ground station nodes must be accounted for and removed from the overall network access time computation.

Overlap between ground station nodes is not necessarily a bad thing. Should two ground station nodes be within the footprint of a cubesat it is true that they will both be attempting to receive, demodulate, and decode the same burst of information. However, if there is large spatial diversity between the nodes, then there is an inherent data protection capability for the transmitted data. With large spatial diversity between nodes comes a decorrelation of the channels between the cubesat and the respective nodes, and thus independence of the channels. Should one of the channels from the cubesat to a given node become corrupted, say for example due to local interference, this corruption is independent of the cubesat's channel to another node. If all of the nodes are designed properly to have a low probability of error, if one node should fail to decode a packet of data due to local problems, this has no influence on another node's probability of properly decoding a packet. This is a very significant effect, and is one of the biggest benefits of a distributed ground station network as it increases the overall probability of success in downloading data over a single cubesat / single ground station design.

4.1.3 Backhaul Connectivity

In order to properly realize a distributed network of ground receiver nodes, it is important that they be able to properly communicate with either each other or with a central server. By the distributed nature of the design of the network, any given specific node will only have a snapshot of part of the pass of a satellite for a given period of time. It is the combination of each nodes contribution of collected data that makes the network beneficial. Therefore, each node must be capable of forwarding data, for its snapshot of a pass, to a central server. The central server can then collectively combine and collate the data from the various nodes of the network. For the purposes of the simulations presented in this section, it is assumed that there is constant connectivity between the receiver node and a central server such that, as data is successfully collected at a node, it can immediately forward that data to the central server. Later, when a candidate receiver node design is presented, the Internet is used as this backhaul network. This connectivity also enables the potential for a number of other features of a distributed ground station network, to include signal combining, and orbit determination.

4.2 STK Simulations

4.2.1 Scenario Constraints

There are multiple ways to simulate a Distributed Ground Station Network (DGSN). For the purposes of this work, AGI's Systems Tool Kit (formerly Satellite Tool Kit, a.k.a STK) is used to model the astrodynamics of the spacecraft and its orbit. Multiple ground station receiver nodes are placed on various points of the Earth. The specific algorithm used by STK to model the spacecraft and its orbit is the SGP4 algorithm, which utilizes the Two Line Element (TLE) set for a spacecraft. Two Line Elements are a dataset that contain all of the orbital elements of a spacecraft necessary to describe the orbit of a spacecraft.

Ground Locations

The DGSN receiver node locations were selected somewhat arbitrarily, with a single purpose, to maximize coverage by increasing the spatial diversity. If all 19 nodes were located in the same state, there would be little benefit of a DGSN because there would be significant overlap of transmitted data to each node. Therefore, the goal was to spread out the receiver node locations such that there is large spatial diversity between them. Table 4.1 gives the latitude, longitude, and elevation of the node locations. Figure 4.1 shows the node locations on STK's 3D viewer (not visible is McMurdo Station, Antarctica). Note how the nodes are roughly distributed evenly throughout North America, with the obvious exception of locations such as Alaska, Hawaii, and Antarctica.

Table 4.1: Network Receiver Node Locations.

#	City_State	Latitude (deg)	Longitude (deg)	Altitude (km)
1	Anchorage_AK	61.175	-149.993	0.031
2	Atlanta_GA	33.748	-84.388	0.313
3	Augusta_ME	44.312	-69.783	0.020
4	Austin_TX	30.267	-97.743	0.305
5	Blacksburg_VA	37.205	-80.417	0.633
6	Denver_CO	40.039	-104.984	1.6475
7	Edmonton_CAN	53.544	-113.491	0.671
8	Helena_MT	46.596	-112.027	1.237
9	Honolulu_HI	21.304	-157.858	0.004
10	McMurdo_ANT	-77.845	-75.597	0.034
11	Miami_FL	25.79	-80.13	0.0024
12	Oklahoma_OK	35.468	-97.516	0.396
13	Olympia_WA	47.037	-122.9	0.031
14	Phoenix_AZ	33.446	-112.077	0.340
15	Pierre_SD	44.368	-100.35	0.442
16	Sacramento_CA	38.582	-121.493	0.009
17	SanJuan_PR	18.451	-66.088	0.003
18	Toronto_CAN	43.653	-79.384	0.076
19	Urbana_IL	40.111	-88.228	0.222

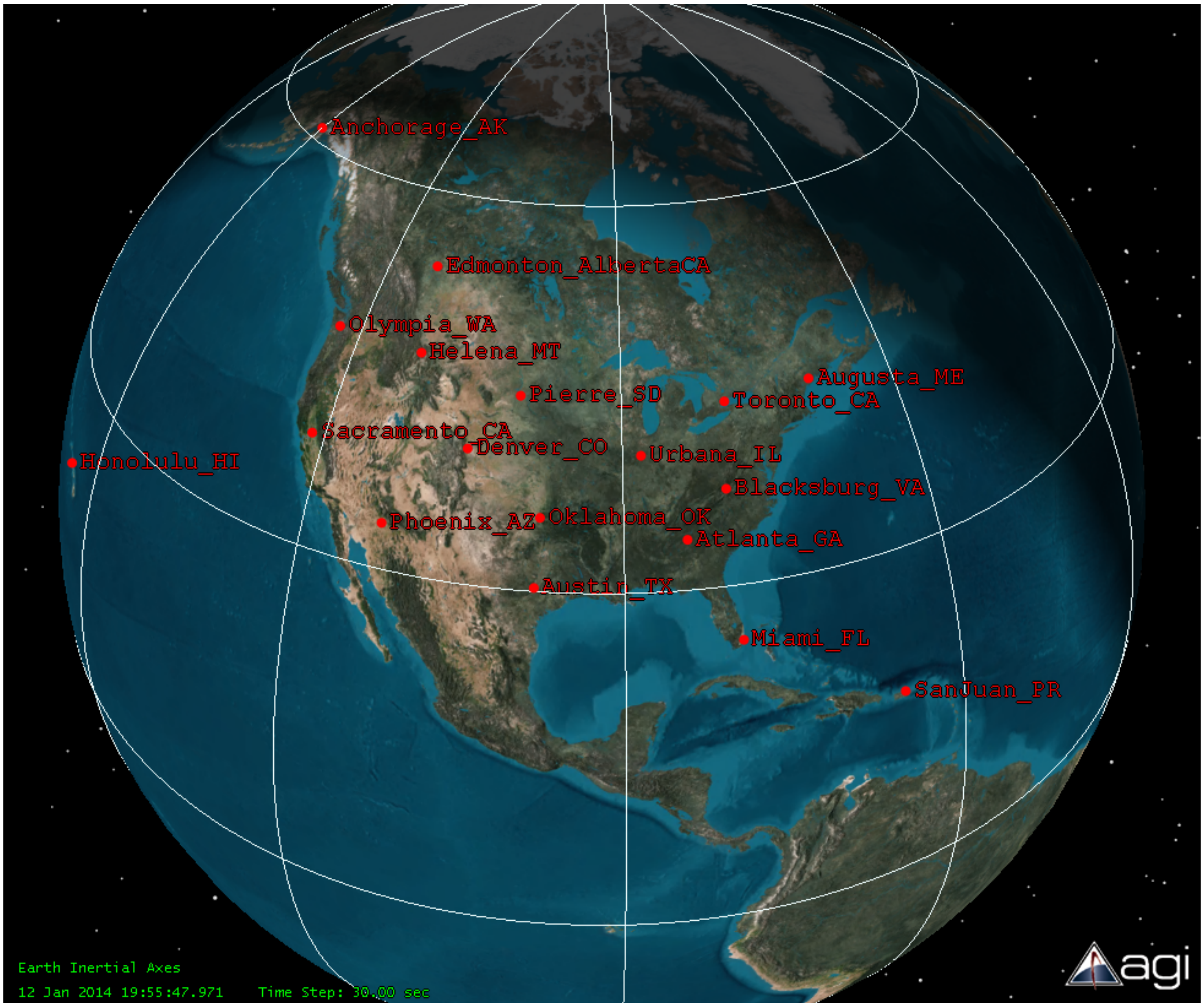


Figure 4.1: Network Receiver Node Locations.

Selected Spacecraft

Table A.1 in Appendix A shows a list of currently active cubesats. The orbits of these 45 spacecraft are fairly diverse. However, there are definite patterns, most notably in orbital altitude and inclination. Four cubesats have been selected for the simulations presented in this section that are representative of the larger set of orbits that can be expected for a Low Earth Orbit deployment of a cubesat. The **sub-satellite point** is defined as the point on the Earth's surface where the line connecting a spacecraft to the center of the Earth crosses through the surface of the Earth. **Orbital altitude** is defined as the distance between the satellite and its sub-satellite point. **Inclination** is defined as the angle of the orbit of the spacecraft with respect to the equator in the direction of Earth's rotation and rotating north.

Table 4.2: Spacecraft Selected for Simulation.

Common Name	NORAD ID	Average Altitude [km]	Inclination [deg]
SWISSCUBE-1	35932	725	98
XATCOBEO	38082	646	70
ISS (ZARYA)	25544	423	52
JUGNU	37839	851	20

The four spacecraft selected for simulation are listed in Table 4.2 with their NORAD Identification Number, average orbital altitude, and inclinations listed. With the exception of Xatcobeo, the spacecraft all have nearly perfect circular orbits. SwissCube has a nearly polar orbit with a desirable orbital altitude. Xatcobeo has a higher ellipticity in its orbit with a perigee (point closest to the earth in the orbit) of about 314 km and an apogee (point furthest from the earth in the orbit) of about 982 km. It is this specific reason that Xatcobeo was selected for simulation, to provide a dataset for an orbit that is *not* nearly circular. The International Space Station, obviously is *not* a cubesat. However, the ISS does deploy cubesats using a mechanism similar to the PPOD, and thus its orbit is representative of a potential orbit that may be achieved by a cubesat. It also has one of the lowest orbital altitudes. Jugnu was selected because of its low inclination angle, making it less accessible by nodes the further they are from the equator.

Table 4.3: Selected Spacecraft Control Facility Locations.

Common Name	Facility	Location	Latitude (deg)	Longitude (deg)	Altitude (km)
SWISSCUBE-1	EPFL	Lausanne, Switzerland	46.5197	6.56515	0.500
XATCOBEO	UoVigo	Vigo, Spain	42.1699	-8.68612	0.000
ISS (ZARYA)	JSC-NASA	Houston, TX	29.5519	-95.0982	0.013
JUGNU	IIT	Kanpur, India	26.5123	80.2339	0.123

Table 4.3 above shows the control facility locations for the four selected spacecraft. The control facility for SwissCube is located at the Ecole Polytechnique Federal de Lausanne (EPFL) in Luasanne, Switzerland. The control facility for Xatcobeo is located at the University of Vigo in Vigo, Spain. Obviously, the International Space Station has a very complex control network comprised of multiple ground stations and the Tracking and Data Relay Satellite System (TDRSS). However, in order to be representative of a cubesat in a similar orbit, the Johnson Space Center (JSC-NASA) in Houston, Texas is selected as the single control facility for the simulation data presented below. The control facility for Jugnu is located at the Indian Institute of Technology (IIT) in Kanpur, India. The locations of the ground station facilities are used to provide the access counts and access times as a baseline for comparison to the ground station network in Section 4.2.2.

Simulation Period

For all simulation results presented below the simulation period is 31 days in duration. Specifically, the beginning of the simulation is January 1st, 2014 at 00:00:00 UTC (midnight). The simulation terminates on February 1st, 2014 at 00:00:00 UTC (midnight).

Communications Protocol

For the simulation results presented below the link layer protocol utilized is the AX.25 protocol. Specifically it is assumed that Unnumbered Information (UI) Frames are utilized. It is also assumed that for every packet transmitted, the entire payload field of the packet is utilized, thus maximizing the protocol efficiency. As described in Section 3.1 the minimum overhead for AX.25 is 7.25% when the entire payload field of the packet is utilized, thus resulting in a protocol efficiency of 92.75%. The term **raw data** will be used in reference to the actual number of transmitted bits whereas the term **payload data** will be used in reference to the actual amount of data from the spacecraft sensor deck that is transmitted to the ground. The raw data will always be greater than the payload data because the raw data includes such entities as the payload header, checksums, and flag bits. Additionally, further overhead is introduced when the forward error correction is utilized.

Forward Error Correction

For the simulation results below, when forward error correction is included in the simulation the scheme will always be the same. The forward error correction scheme for all simulations that include FEC will be for convolutional coding with rate R=$\frac{1}{2}$ and depth K=7. It is assumed that soft decision decoding is used on the receive side. This is just one type of forward error correction and there are many other options (e.g. Reed Solomon codes). However, the point of the simulation results is not to show every possible scenario, but instead to highlight the differences between three specific types of scenarios: the current type of communications in use, the impact of switching to digital modulation, and the impact of including forward error correction.

Simulation Terminology

Access time will be defined as the amount of time that line of sight exists between the satellite and a ground station node and is counted in seconds. An **Access Window**, **Pass**, or simply an **Access** is very similar to access time, but specifically refers to a moment in time (date and time) for which access time is greater than zero. For example an *access window* may exist from January 1, 2014 00:00:00 UTC to January 1, 2014 00:10:00 and the *access time* is +600 seconds. **Acquisition of Signal (AOS)** will be defined as the moment in time when line of site is acquired between the satellite and a specific ground station node as the satellite is rising, i.e the moment in time that an Access Window begins. **Loss of Signal (LOS)** will be defined as the moment in time when the line of sight is lost with a satellite from a specific ground station node as the satellite is setting, i.e. the moment in time that an Access Window stops. For the following simulation results, there is a ten degree minimum elevation requirement. This means that AOS is not achieved until the satellite has risen ten degrees above the visible horizon as viewed from a specific node. Similarly, the stop of an access window, or LOS, occurs as the satellite has set below ten degrees above the visible horizon.

Network Access time is the amount of time that a satellite is visible to the ground station network and is counted in seconds. Network access time implies that any overlap in time between one node and any other node has been removed and thus represents the amount of continuous time the satellite has line of site with one or more nodes. A **Network Access Window**, **Network Pass**, or simply a **Network Access** refers to the specific moment in time for which network access time is greater than zero. **Network Acquisition of Signal (NAOS)** is the moment in time the satellite gains line of sight to the first ground station node during a network access window. **Network Loss of Signal (NLOS)** is the moment in time the satellite loses line of sight to the last ground station node during a network access window. Again both NAOS and NLOS adhere to the 10 degree elevation requirement.

4.2.2 Simulation Results

Single Ground Station Simulation Results

Before presenting the results for the complete network, it is important to get an idea of the current methods for communicating with cubesats and the expected access times. Figure 4.2 below shows the indivdual count of accesses for each ground station location. Again this is simply a count of how many times each ground station node has an access window with access time greater than or equal to one second. As expected, nodes that are located farther from the equator, such as McMurdo Station, Antarctica and Anchorage, Alaska, have zero access windows with Jugnu due to the low inclination angle of the spacecraft. On the other hand, SwissCube's 98 degree polar orbit makes contact with McMurdo station on nearly every orbit, and thus the access window count is very high.

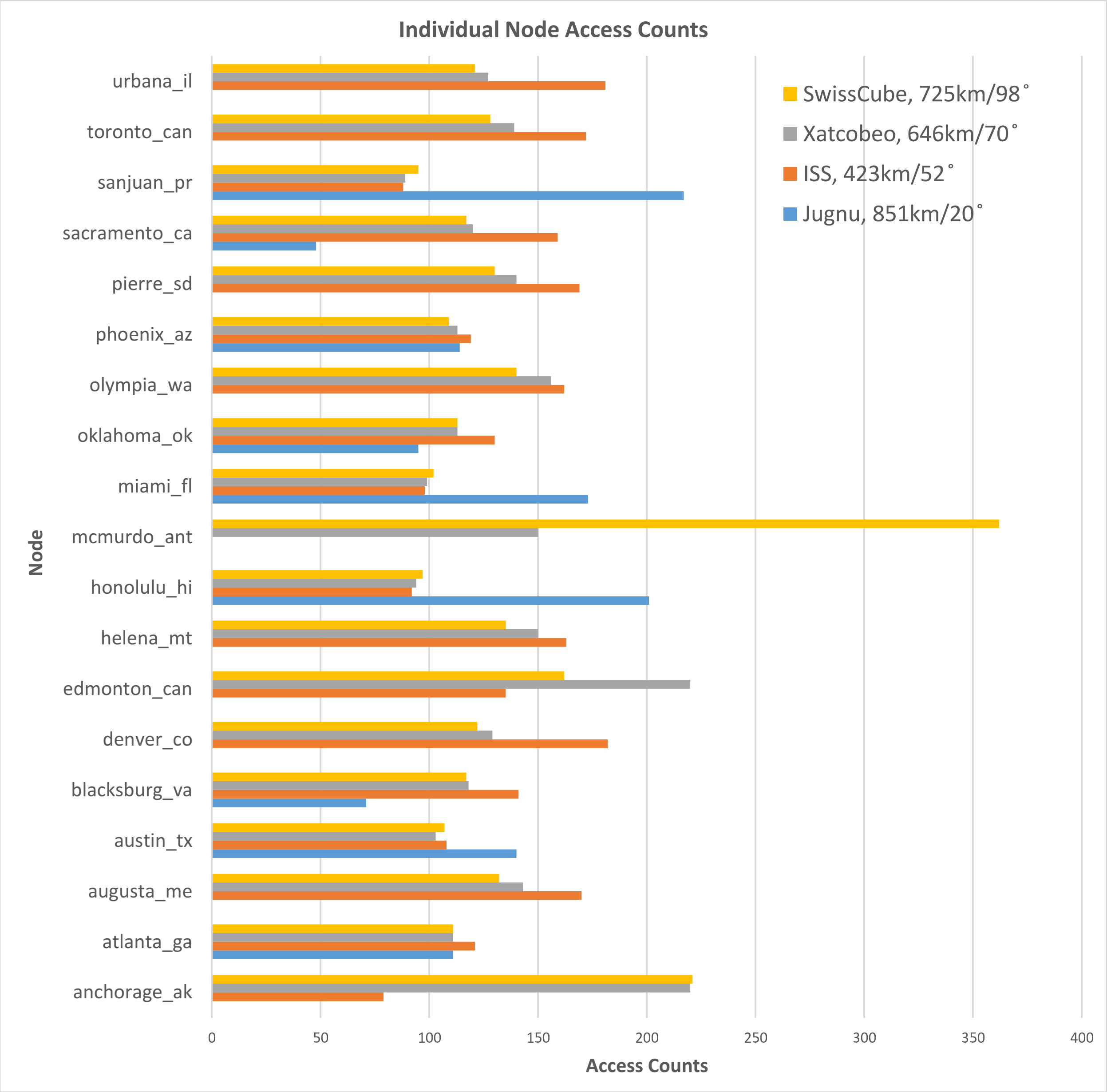


Figure 4.2: Individual Node Access Window Counts.

Figures 4.4 through 4.6 show the maximum expected data volumes per day for each node location. This is computed by taking the total access time during the one month simulation for each ground station node and applying the following equation:

$$V = \frac{(t)(D_R)(R_b)(\eta)}{d} \tag{4.1}$$

Where:

$$
\begin{aligned}
V &= \text{Data Volume, } \left[\frac{\text{bits}}{\text{day}}\right]. \\
t &= \text{Access Time to Spacecraft, [s].} \\
D_R &= \text{Downlink Ratio, fractional \% of time the S/C downlink active.} \\
R_b &= \text{Raw downlink bit rate in bits per second, [bps].} \\
\eta &= \text{Link Layer Protocol Efficiency, (0.9275 for AX.25 UI)} \\
d &= \text{Simulation duration in days, [days].}
\end{aligned}
$$

These plots represent the total amount of payload data that *could* be downloaded at each node location. Stated differently, these plots show the *maximum* amount of data that could be generated per day on the spacecraft and then downloaded to the ground. An important note should be made about downlink ratios. A downlink ratio of 1.0 means that the spacecraft is transmitting during the entire access window. In practice this does *not* happen for a number of reasons. However, a downlink ratio of 1.0 is shown in order to give an absolute upper limit for the possible data volume per day that could be generated by a spacecraft.

Practically speaking, the actual downlink ratio used will vary from spacecraft to spacecraft depending on the decisions of the ground operations team and the design of the spacecraft. As an example, the power budget for the spacecraft may prohibit continuous transmission from the spacecraft. Perhaps atmospheric data is being collected by the spacecraft's sensor suite during a pass over the downlink facility and the design of the spacecraft prohibits collection of data while simultaneously transmitting on the downlink. The actual factors that influence these design considerations and operational decisions are outside the scope of this work, however data volumes were calculated for downlink ratios of 1.0 (maximum upper limit), 0.75, 0.5, 0.25, and 0.1.

Another important note concerns the data itself. The simulations below show actual **payload** data volumes. These data volumes take into account such factors as the AX.25 UI frame efficiency. Additionally, the vertical axes of the plots show two sets of numbers. This is because data volumes were calculated for two possible downlink bit rates (R_b). Specifically, along the left vertical axis, values are given for the 1200 bps raw bit rate commonly associated with the AFSK_FM modulation scheme. Along the right vertical axis values are given for the 9600 bps raw bit rate commonly associated with the G3RUH_FSK modulation scheme.

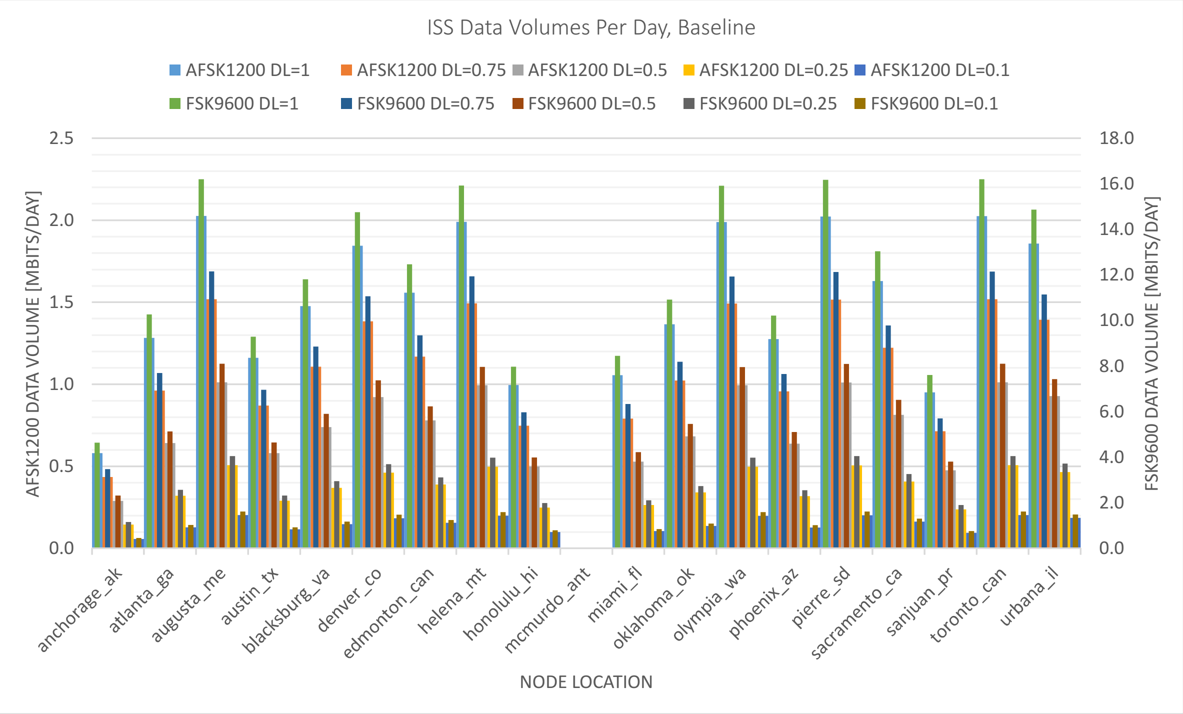


Figure 4.3: ISS Data Volumes.

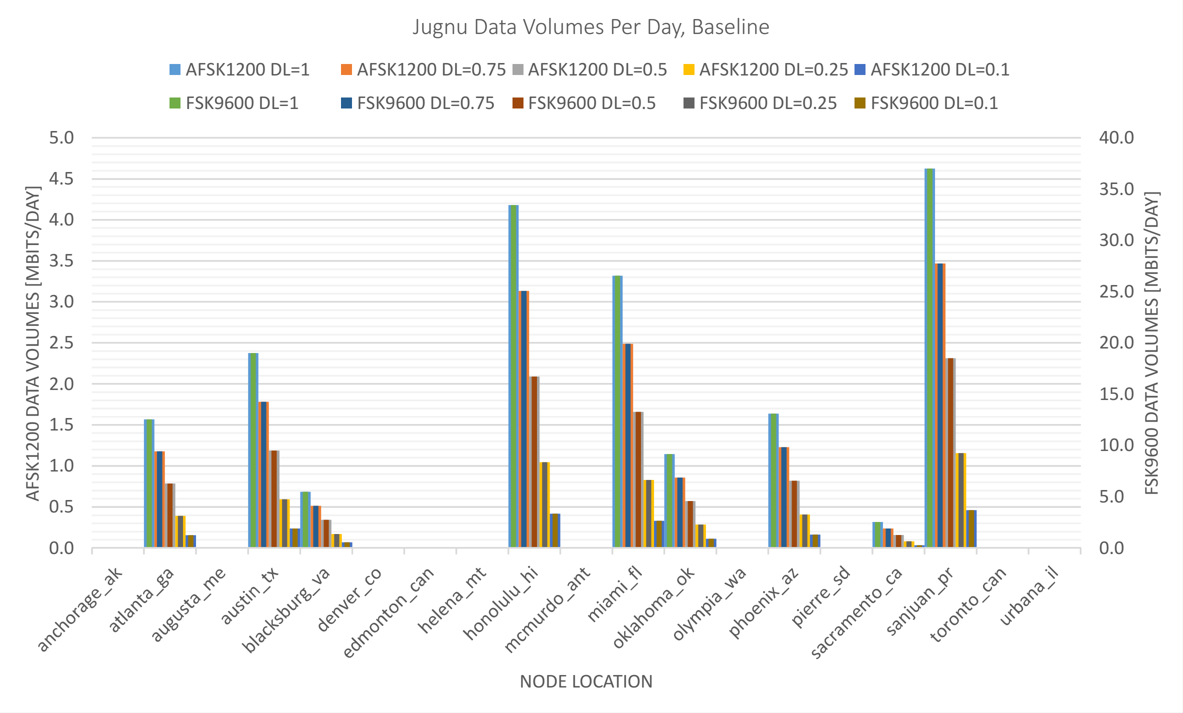


Figure 4.4: Jugnu Data Volumes.

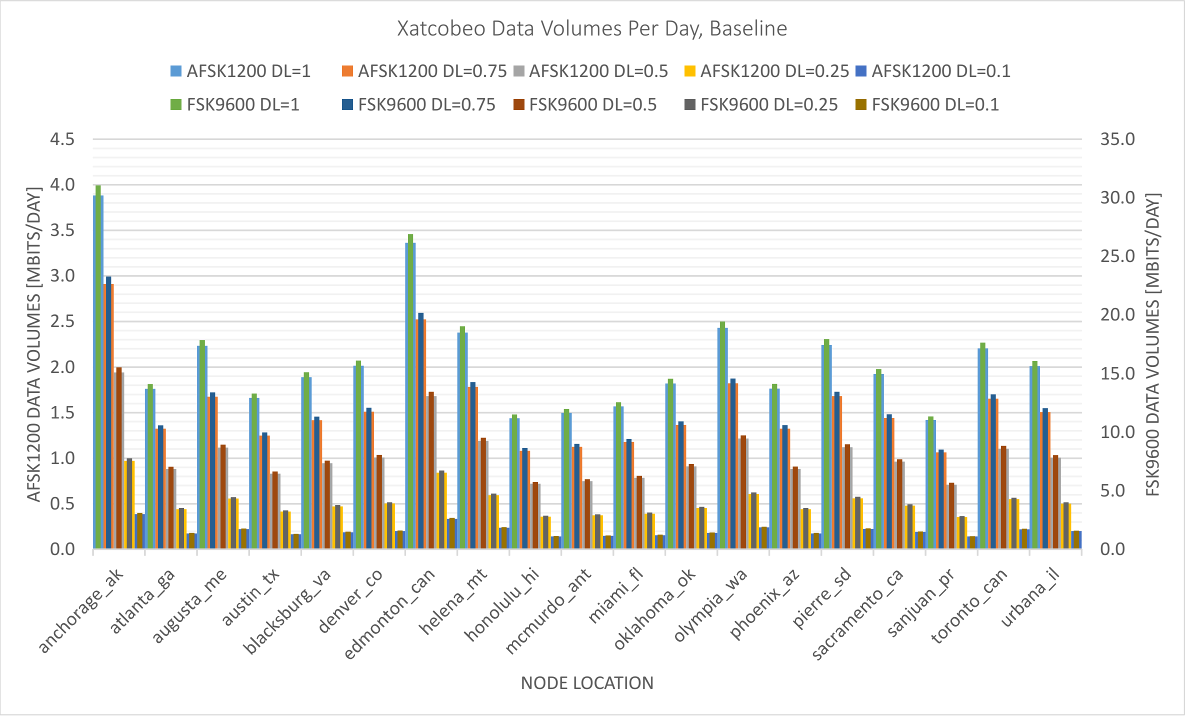


Figure 4.5: Xatcobeo Data Volumes.

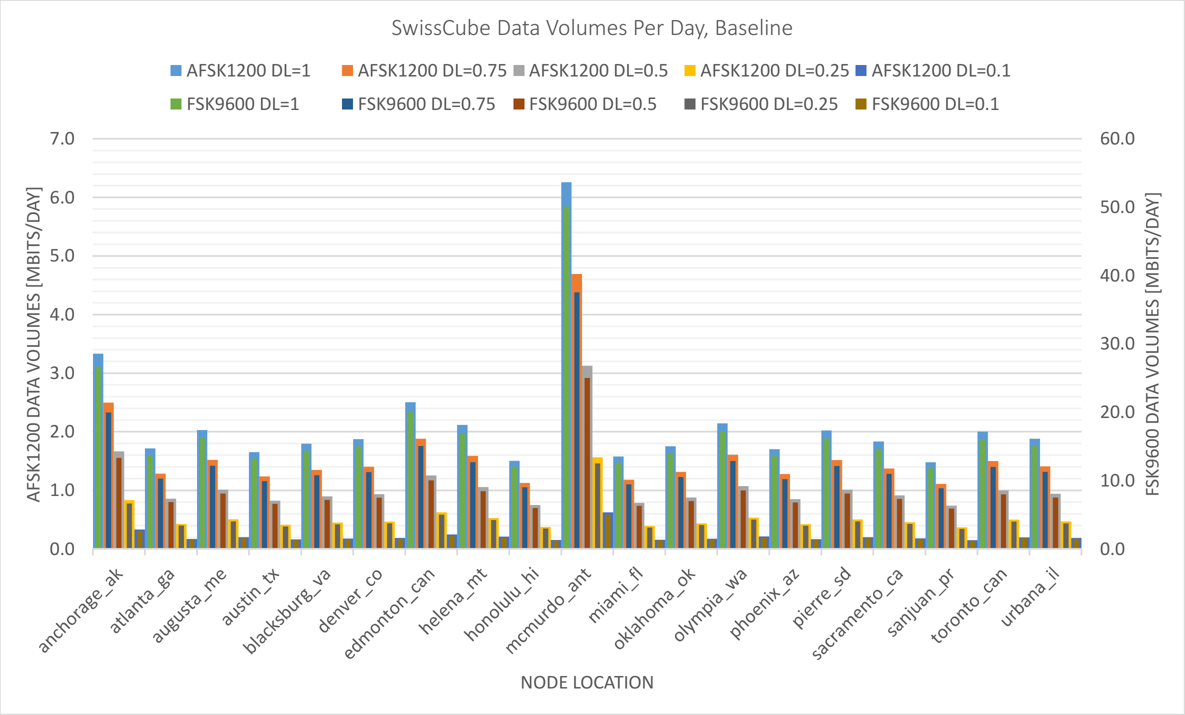


Figure 4.6: SwissCube Data Volumes.

Figure 4.3 above shows the volume of data that can be expected for a spacecraft with an orbit similar to that of the International Space Station. As expected there are zero contacts and thus no data transfer with McMurdo Station, Antarctica. Figure 4.4 above shows the volume of data that can be expected for a spacecraft with an orbit similar to that of Jugnu with a low inclination angle. Again as expected there is no contact and thus no data transfer with ground station nodes located far from the equator. Figure 4.5 above shows the volume of data that can be expected for a spacecraft with an orbit similar to that of Xatcobeo. Notably, Anchorage, Alaska and Edmonton, in Alberta Canada, have much higher data volume transfers. This is because apogee for the orbit is over the Northern Hemisphere and thus access windows are longer for spacecraft with higher latitudes, thus allowing for larger data transfers. Figure 4.6 above shows the volume of data that can be expected for a spacecraft with an orbit similar to that of SwissCube. Due to the polar nature of SwissCube's orbit, access windows exist with McMurdo Station, Antarctica on almost every orbit, thus allowing for large amounts of data transfer, whereas the rest of the ground station nodes have roughly equal contact times.

Distributed Ground Station Network Simulation Results

Again all of the simulation results so far show what can be expected for single ground station node accesses to each of the four spacecraft. The following simulation results, for the same four spacecraft show the effect of combining the 19 individual ground station nodes into a distributed network operating in concert. In addition to combining the nodes into a network, the effect of switching from the analog communications schemes to the digital schemes, as discussed in Chapter 3, is also shown in the results below. There are two approaches for the following analysis that can be taken.

The first type of analysis is when the data volume for a particular spacecraft is known. Given this information, combined with the expected amount of network access time (as a function of the orbit and the geometry of the ground station network), the required downlink bit rate can be determined. This would have the benefit of allowing the bandwidth of the signal to be adjusted according to the specific requirements of the spacecraft. However, this work is not specific to any one spacecraft. Therefore, this type of analysis will not be performed. However, it is important to highlight the utility of this calculation.

Instead, the second approach is taken, similar to the single ground station results above. The total data volume per day that can be generated onboard the spacecraft is determined for the simulation results presented below. Again, a number of assumptions are made. As in the single ground station results above, it is assumed that the link layer protocol used is AX.25, and thus an efficiency of 92.75%. Also, results are presented for various downlink ratios. Again, a downlink ratio of 1.0 is impractical, but is shown to give an absolute upper limit in the data volume per day calculation. Finally, the maximum channel bandwidth of 20 kHz is assumed to be occupied.

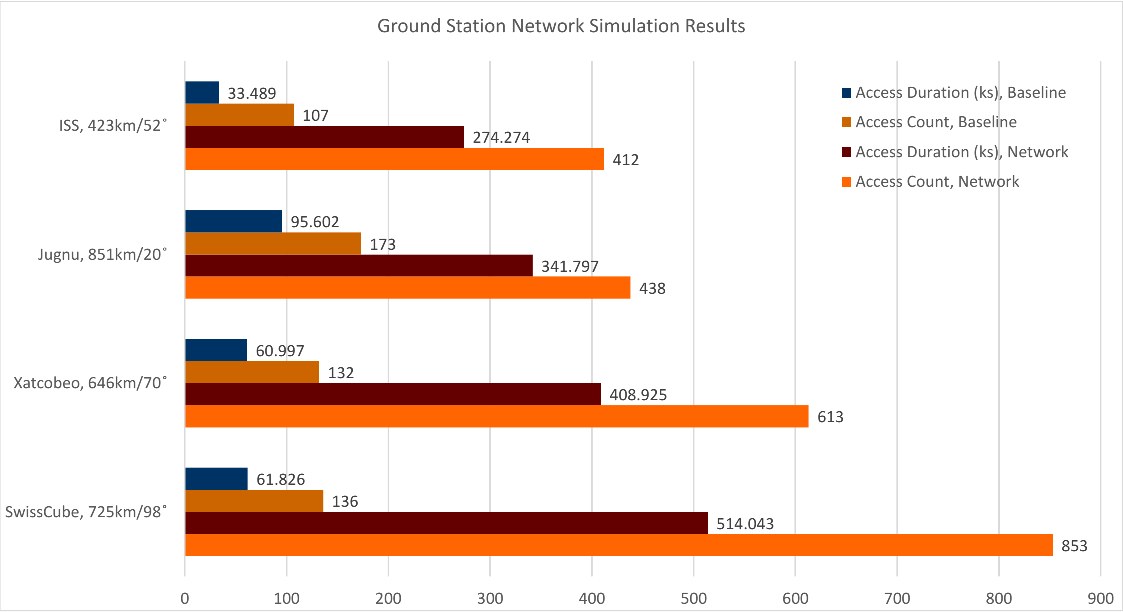
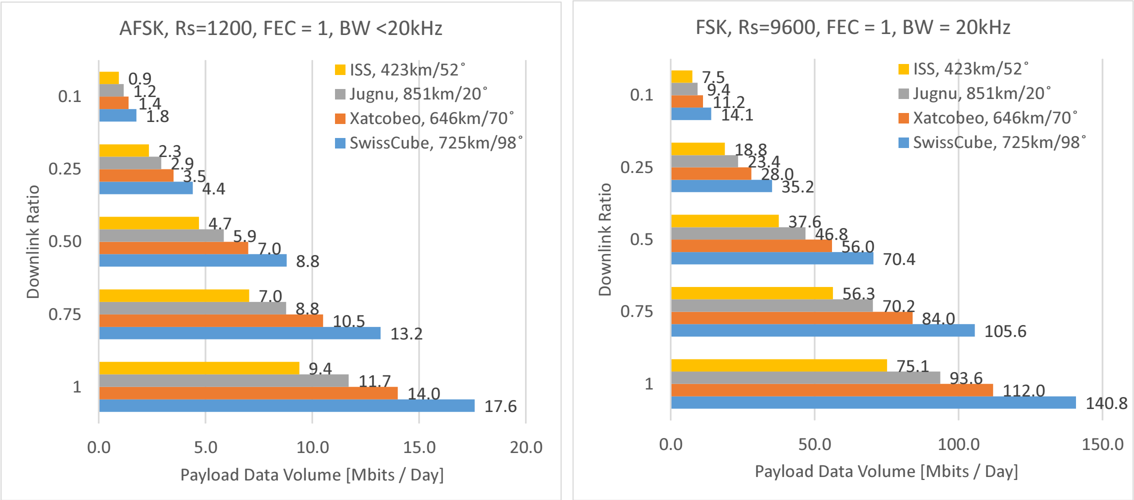


Figure 4.7: Network Simulation Results.

Figure 4.7 shows the count of network access windows, as well as the total network access time for the four spacecraft. Note that the access time shown is in *kilo*-seconds. As a baseline, both the access duration and access count for the individual spacecraft's designated control facility is shown. The control facility locations are listed in Table 4.3 and the simulation period is the same as the network simulation period. For the ISS, the ground station network offers 8.2 times the amount of access time over the baseline access time for the Johnson Space Center. For Jugnu, the ground station network offers 3.6 times the amount of access time over the baseline access time for the IIT. For Xatcobeo, the ground station network offers 6.7 times the amount of access time over the baseline access time for the University of Vigo. For SwissCube, the ground station network offers 8.3 times the amount of access time over the baseline access time for EPFL. The impact of this, as translated to data volumes, is readily apparent and shows the significance gains of a distributed ground station network model over the single control facility model.

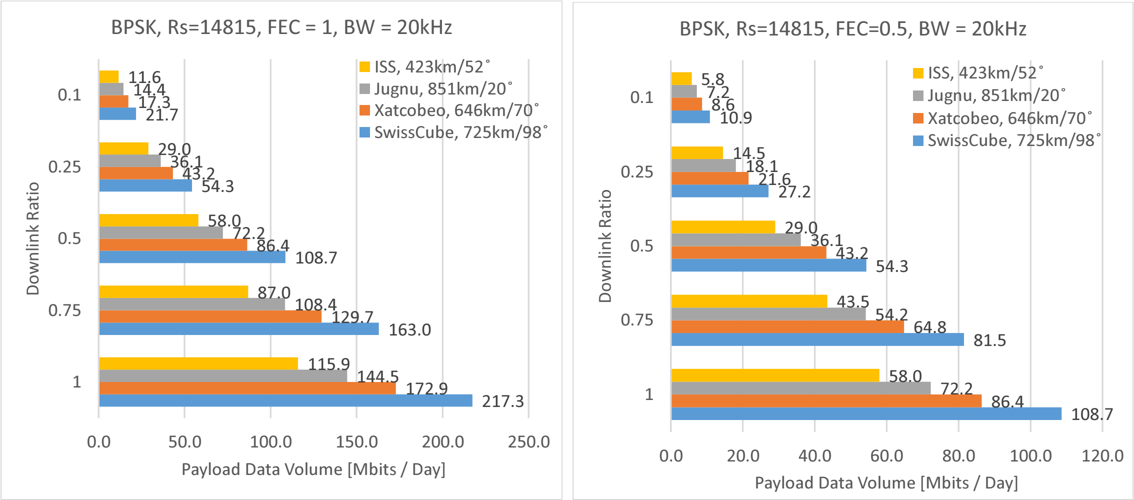
It is the *network access time* shown in Figure 4.7, for each spacecraft respectively, that is used in the following network simulation results. Again as a baseline, the per day volume of data is calculated for AFSK1200 and FSK9600 schemes, the common schemes currently in use. This shows the amount of data a spacecraft could generate per day that could be downloaded to the distributed ground station network using the two most common communications protocols in use by existing cube satellites. The results are shown if Figure 4.8 below.



(a) AFSK 1200, AX.25, Full Network. (b) FSK 9600, AX.25, Full Network.

Figure 4.8: Full Network, Baseline, Data Volume Capacity.

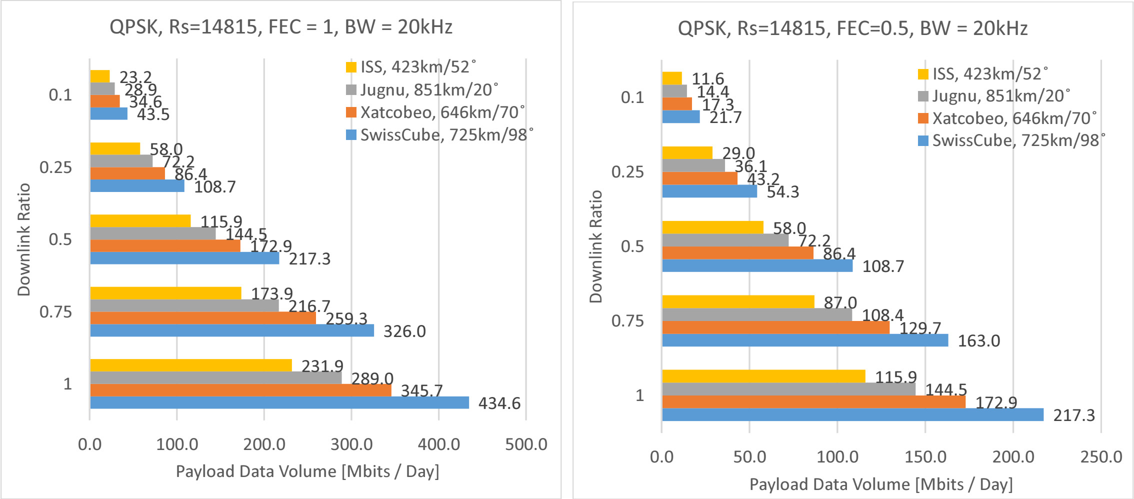
The next set of figures shows the per day volume of data that could be received by the network for BPSK modulation. For this set of results it is assumed the maximum RF Bandwidth of 20 kHz is occupied and the excess bandwidth factor is 0.35. This results in a symbol rate of 14.815 ksps, and with 1 bit per symbol for BPSK, we have a downlink data rate of 14.815 kbps. Figure 4.9a shows the results with no forward error correction. Figure 4.9b shows the results with rate R=$\frac{1}{2}$ Convolutional Coding.



(a) BPSK, Full Rate, No FEC. (b) BPSK, Full Rate, FEC=$\frac{1}{2}$.

Figure 4.9: Full Network, BPSK, Data Volume Capacity.

The next set of figures shows the data volume per day of data that could be received by the network for QPSK modulation. For this set of results it is assumed the maximum RF Bandwidth of 20 kHz is occupied and the excess bandwidth factor is 0.35. This results in a symbol rate of 14.815 ksps, and with 2 bits per symbol for QPSK, we have a downlink data rate of 29.630 kbps. Figure 4.10a shows the results with no forward error correction. Figure 4.10b shows the results with rate R=$\frac{1}{2}$ Convolutional Coding.



(a) QPSK, Full Rate, No FEC. (b) QPSK, Full Rate, FEC=$\frac{1}{2}$.

Figure 4.10: Full Network, QPSK, Data Volume Capacity.

Chapter 5

Receiver Node Design

5.1 System Overview

The heart of the design of the distributed ground station network node involves the selection of two key parts, a software defined radio (SDR) receiver and an embedded computer, a.k.a. a General Purpose Processor (GPP), capable of processing signals. All other components are developed around the SDR and GPP in an effort to make a robust, remotely controllable, and sensitive RF receiver capable of tracking and receiving signals from cubesats in Low Earth Orbit. A general system block diagram is shown in Figure 5.1 below.

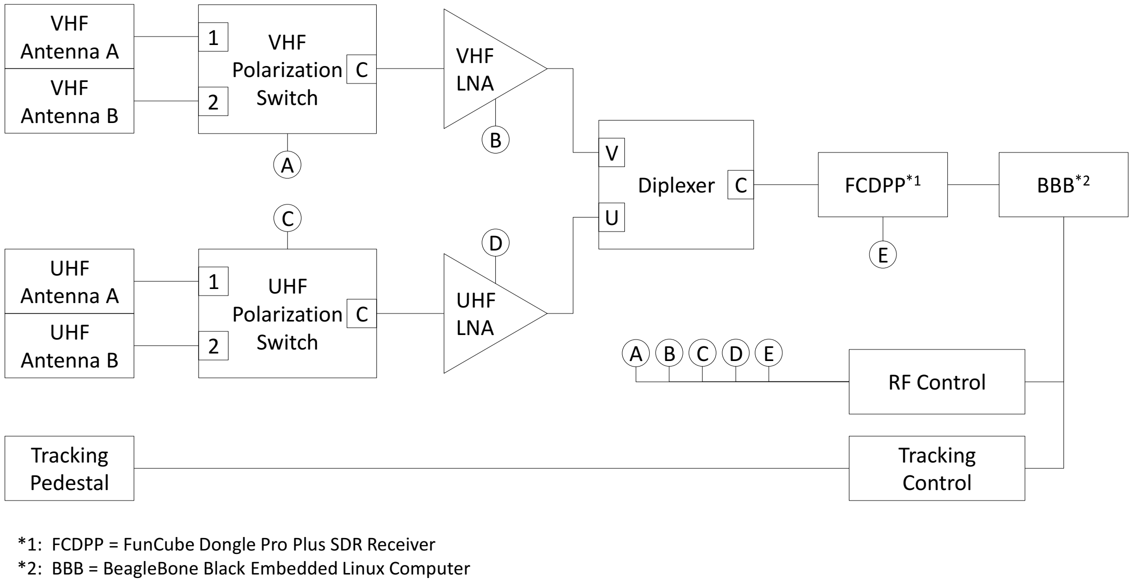


Figure 5.1: System Block Diagram.

5.1.1 SDR Receiver

The selection of Software Defined Radio technology over a standard hardware radio is primarily due to the need for flexibility. Instead of building multiple hardware demodulators for existing cubesat protocols and then having to physically upgrade the system as new protocols become implemented on new spacecraft, this can now all be accomplished in software. Waveforms (applications that process radio signals in the software domain) can be written for specific spacecraft which conform to that spacecraft's specific modulation scheme, FEC scheme (if used), and higher layer communication protocol. Instead of having to physically travel to each node and install a new piece of hardware, the new waveform can simply be downloaded over the Internet to the receiver nodes, and then executed when there is a pass of the spacecraft.

The SDR receiver selected for this project is the FunCube Dongle Pro Plus (FCDPP) from Howard Long, G6LVB [15]. A number of candidate SDR systems were investigated, but for one reason or another were discarded. These SDRs include the RTL-SDR, the HackRF Jawbreaker, and the Universal Software Radio Peripheral Embedded Series from Ettus Research. Criteria for selection included receiver noise performance, interface methods, and cost. The FCDPP was ultimately selected due to a balance of cost ($209.11 USD), a simple interface method (USB), excellent noise performance (more details in Section 5.1.4), and the specific attention in the design of this SDR towards its application for reception of LEO cubesat signals.

The FCDPP appears as a soundcard device to the host computer system. It passes the 16 bit In-phase and Quadrature (IQ) samples utilizing the 16bit, little endian, PCM interface of the soundcard to the host, with the In-phase data transferred on one stereo channel and the Quadrature data transferred in the other stereo channel. The sample rate of the FCDPP is fixed at 192 kilo-samples per second (ksps). Because it is capturing both I and Q data at a rate of 192 ksps it is capable of capturing an instantaneous 192 kHz of RF Spectrum (± 96kHz above and below the tuned RF center frequency). Finally, the FCDPP has a Noise Figure of 3.5 dB in both the 2m and 70cm Amateur Radio bands[15]. More details about the control and signal processing software are given in Section 5.1.3.

5.1.2 Embedded Computer

Initially, the embedded computer selected for this design was the RaspberryPi Model-B rev 1 [16]. This system has a 700 MHz ARMv6 processor with floating point emulation, 512 MB of RAM, USB connectivity, 100 MB Ethernet, and is capable of running a full Linux OS distribution. This embedded computer has an extremely low cost of $40.00 USD. However, the lack of a floating point co-processor, and the need to utilize floating point emulation for the necessary signal processing results in a system that simply does not have enough processing horsepower for SDR work. The author's only successful experiment with the combination of the Raspberry Pi and the FCDPP was a small application written in C that was able to stream raw samples from the FCDPP, via TCP/IP, to a host computer running an SDR waveform (GNU Radio). Unfortunately, this is not sufficient for the goals of this project. GNU Radio was successfully installed on the Raspberry Pi in an attempt to process signals from the FCDPP onboard the Raspberry Pi, however even the simplest GNU Radio waveforms caused the processor to bog down entirely when executed. After initial testing with the Raspberry Pi failed, the BeagleBone Black was tested next.

The final embedded computer selection for this design is the BeagleBone Black (BBB)[17]. The BBB has an AM335x 1.0 GHz ARM Cortex-A8 processor with a NEON floating point co-processor. From a processing power point of view this is sufficient for all necessary signal processing (with room to spare). In addition to its ability to process signals, the BBB also comes with a number of features needed for this system, including 512 MB RAM, a USB host interface (for connection to the FCDPP), 100 MB ethernet (for connection to the Ground Station Network via the Internet), and an external connector that carries up to 4 UART signals and a number of GPIO signals. Additionally, there are 2 GB of onboard flash storage with a high write speed (compared to the μSD card) which will be useful for storing captured samples. It is currently running a full distribution of Ubuntu Linux v12.04 LTS installed on a 16 GB μSD Card. GNU Radio has also been installed on this system and a number of waveforms have been successfully executed onboard the BBB.

5.1.3 Software

A number of pieces of software are required to realize the design of this system, and not all of them have been completed at the time of this writing. At the core of the design is the Software Defined Radio technology. The SDR Architecture chosen for this design is GNU Radio [18]. GNU Radio is a collection of signal processing blocks written in C and C++. Python is used to connect the various processing blocks and launch the waveforms, making it very easy to develop using GNU Radio.

Control of the FCDPP for the purposes of turning the internal LNA on and off as well as setting the center frequency of the device was accomplished using a small command line application called *fcdcontrol* written by David Pello, EA1IDZ. The detailed instructions for the installation and use of this utility can be found in [19]. The GNU Radio blocks mentioned in the previous paragraph as well the *fcdcontrol* were encapsulated in a GNU Radio Hierarchical Block (a hierarchical block is a GNU Radio block made up of sub blocks) for use with the previously mentioned waveforms. More details are given in Section 5.3.

The last major piece of software utilized by the system is the `predict` satellite tracking software written by John A. Magliacane, KD2BD. This software provides pass prediction capabilities (utilizing TLEs and the SGP4 algorithm) as well as tracking features for use in the receiver node. The benefit of this particular application is the UDP server feature which allows for third party software to be written that interfaces with the orbital tracking program. A simple Python utility can interface with the `predict` application in order to automatically steer the antennas to track the spacecraft.

5.1.4 Noise Performance

Before going into the details of the hardware design of the ground station network receiver nodes, it is important to determine the RF noise performance of the system to ensure it will be capable of receiving signals from spacecraft in Low Earth Orbit. In this section the overall system noise performance of the node is determined. With this information one can then perform a link budget analysis for potential target spacecraft to determine whether or not the link can be closed and the signals can be demodulated.

The key equation used to determine the overall system noise performance is the cascaded noise factor equation (Equation 5.1 below from [20]). This equation takes into account the gain and noise factor of each component of the RF chain, excluding the antenna itself. Once the cascaded noise factor is determined for all of the components, the value is converted to a component noise temperature and added to the antenna noise temperature in order to determine the overall *System* Noise Temperature. All variables in Equation 5.1 shown below are in linear form and must be converted from the logarithmic (dB) scale before the computation is performed. The lower case nf indicates a linear term known as *noise factor*, whereas the capital NF is the logarithmic scaled *noise figure* (Equation 5.5 shows the conversion equation between the two).

$$nf_{cascaded} = nf_1 + \sum_{i=2}^{N} \frac{nf_i - 1}{\prod_{j=1}^{i-1} g_j} = nf_1 + \frac{nf_2 - 1}{g_1} + \frac{nf_3 - 1}{g_1 g_2} + \frac{nf_4 - 1}{g_1 g_2 g_3} + \dots \quad (5.1)$$

In addition to the cascaded noise factor equation (Equation 5.1) a number of equations will be required in order to determine the overall system noise temperature. Equation 5.2 shows the conversion from the linear noise factor to noise temperature. In order to determine the system noise temperature the antenna noise temperature is added to the cascaded noise temperature as shown in Equation 5.3. For the antenna noise temperature a value of 290 Kelvin [K] is assumed, which is the widely accepted noise temperature generated by the Earth. Once the overall system noise temperature is determined it is used as part of the computation for determining the Carrier to Noise Ratio of a link. The noise temperature calculation is shown in this section and the actual link budget analyses that make use of the results of this section are shown in Chapter 6.

$$
\begin{aligned}
T &= 290 * (nf - 1) && (5.2)\\
T_{sys} &= T_{comp} + T_{ant} && (5.3)\\
nf_{sys} &= \frac{T_{sys}}{290} + 1 && (5.4)\\
NF_{sys} &= 10 * \log_{10}(nf_{sys}) && (5.5)
\end{aligned}
$$

There are multiple components in the RF chain that need to be taken into account for the Noise Figure computation from the antenna all the way to the SDR receiver. Additionally, the ground station receiver node is designed to operate in the 2m and 70cm Amateur Radio bands. Therefore, two system noise temperatures must be computed, one for each band. The 2m component noise characteristics are shown in Table 5.1 and the 70cm component noise characteristics are shown in Table 5.2 below. There is a double hit with the passive devices, Insertion Loss (which can be treated as negative gain) and increased Noise Figure that is equivalent in value to the Insertion Loss.

Table 5.1: 2m Component Noise Characteristics.

#	**Component**	**IL[dB]**	**G[dB]**	**g[linear]**	**NF[dB]**	**nf[linear]**
1	BNC(M)-N(F) Adapter	0.070	-0.070	0.984	0.070	1.016
2	LMR-400UF (6ft)	0.102	-0.102	0.977	0.102	1.023
3	N(F)-SMA(M) Adapter	0.070	-0.070	0.984	0.070	1.016
4	FLEX141, (9in)	0.030	-0.030	0.993	0.030	1.007
5	Pol SW, Coax Relay	0.100	-0.100	0.977	0.100	1.023
6	UT141, (10in)	0.033	-0.033	0.992	0.033	1.008
7	VHF LNA, AR^2 P144VDG	N/A	24.000	251.189	0.500	1.122
8	UT141, (10in)	0.033	-0.033	0.992	0.033	1.008
9	Diplexer, 2m Branch	0.380	-0.380	0.916	0.380	1.091
10	SMA(M)-N(F) Adapter	0.031	-0.031	0.993	0.939	1.241
11	LMR-400UF, (4ft)	0.068	-0.068	0.984	0.068	1.016
12	SMA(M)-N(F) Pigtail	0.031	-0.031	0.993	0.031	1.007
13	FunCube Dongle Pro+		10.000	10.000	3.500	2.239

Table 5.2: 70cm Component Noise Characteristics.

#	**Component**	**IL[dB]**	**G[dB]**	**g[linear]**	**NF[dB]**	**nf[linear]**
1	BNC(M)-N(F) Adapter	0.070	-0.070	0.984	0.070	1.016
2	LMR-400UF, (6ft)	0.186	-0.186	0.958	0.186	1.044
3	N(F)-SMA(F) Adapter	0.070	-0.070	0.984	0.070	1.016
4	FLEX141, (9in)	0.068	-0.068	0.985	0.068	1.016
5	Pol SW, Coax Relay	0.100	-0.100	0.977	0.100	1.023
6	UT141, (10in)	0.075	-0.075	0.983	0.075	1.017
7	UHF LNA, AR^2 P432VDG	N/A	17.000	50.119	0.500	1.122
8	UT141, (10in)	0.075	-0.075	0.983	0.075	1.017
9	Diplexer, 70cm Branch	0.550	-0.550	0.881	0.550	1.135
10	SMA(M)-N(F) Pigtail	0.053	-0.053	0.988	0.939	1.241
11	LMR-400UF, (4ft)	0.124	-0.124	0.972	0.124	1.029
12	SMA(M)-N(F) Pigtail	0.053	-0.053	0.988	0.053	1.012
13	FunCube Dongle Pro+		10.000	10.000	3.500	2.239

Applying Equation 5.1 for each band to determine noise factor (linear scale) we have:

$$\begin{aligned} nf_{2m} &= 1.2395[linear] \\ nf_{70cm} &= 1.3242[linear] \end{aligned}$$

We then apply Equations 5.2 to determine the component noise temperature. Then, to determine the overall system noise temperature we apply Equation 5.3, remembering to use the assumed value of 290 [K] for the antenna noise temperature. The resulting overall system noise temperatures, which are quite acceptable, are shown below.

$$\begin{aligned} T_{sys\_2m} &= 359.46[K] \\ T_{sys\_70cm} &= 384.02[K] \end{aligned}$$

Again, these system noise temperatures are used to perform the link budget analyses presented in Chapter 6.

5.2 Receiver Node Hardware Design

This section will describe the hardware design of the ground station receiver node. Figures 5.2 and 5.3 show the completed prototype. The subsystems in the figure are described in the following subsections.

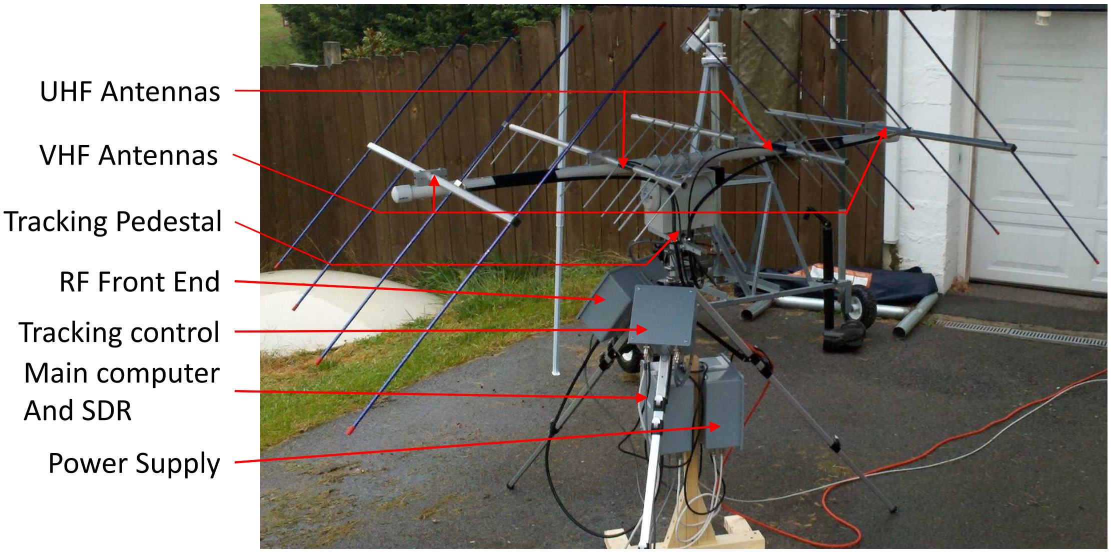


Figure 5.2: Completed System Prototype.

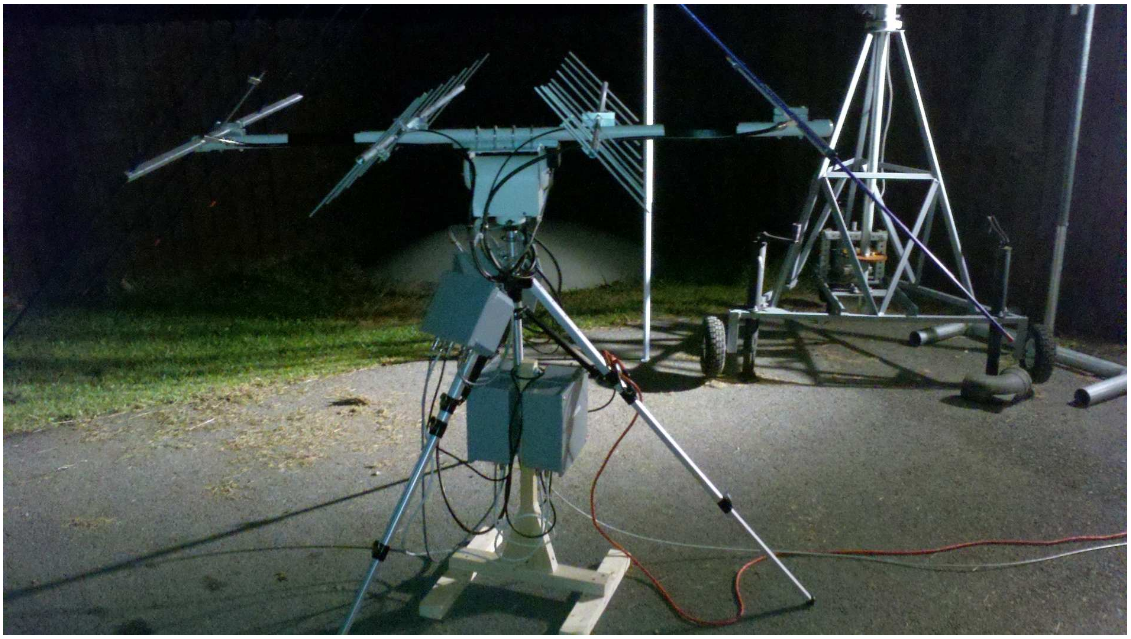


Figure 5.3: Completed System Prototype, alternate view.

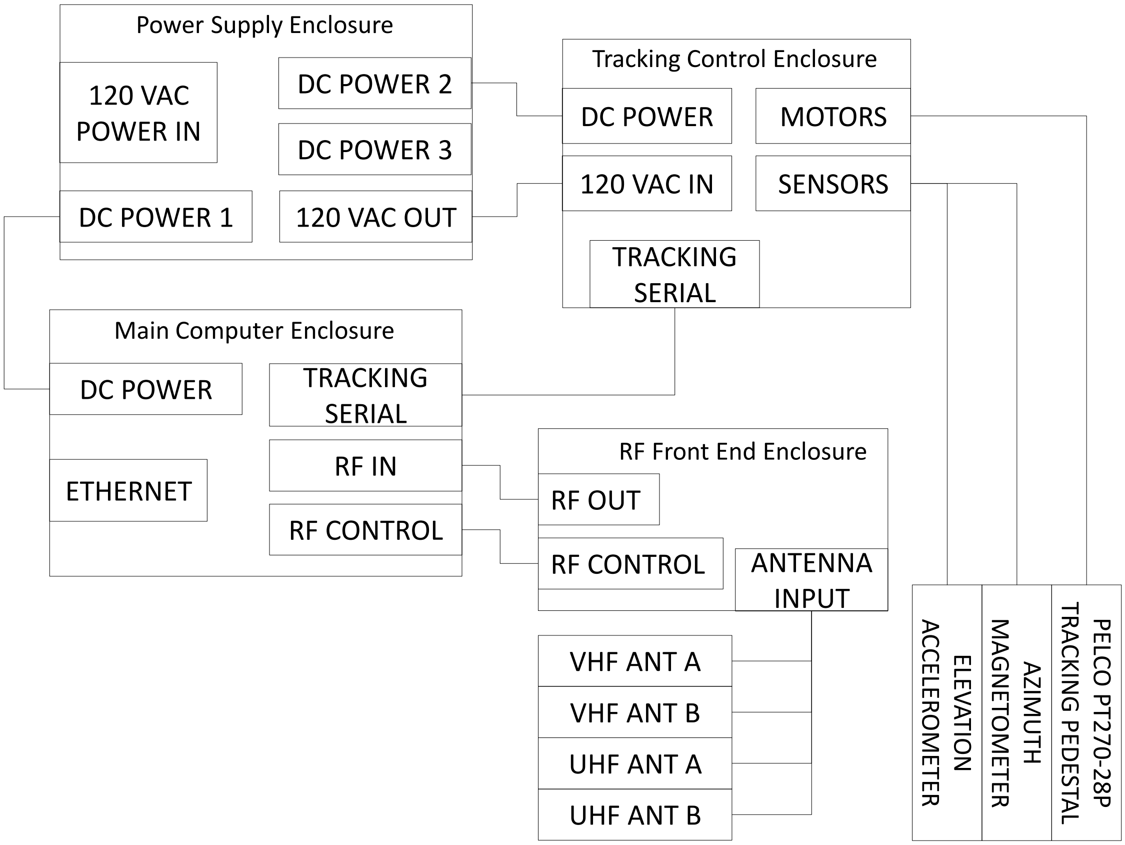


Figure 5.4: Subsystem Interconnection.

As depicted in Figure 5.1, the ground station network receiver node design consists of RF interconnections, control, and power signals. For the physical implementation of the system, five subsystems have been built to realize the design. Each subsystem is housed in its own external enclosure to make the system somewhat modular to allow for future improvements without having to rebuild the entire system. These subsystems include the Main Computer and Radio Subsystem, The Antenna Subsystem, The RF Front End Subsystem, the Tracking Control Subsystem, and the Power Supply Subsystem. The following section breaks down the design of each subsystem. Figure 5.4 shows the interconnection of the subsystems' External Enclosures.

5.2.1 Main Computer and Radio Subsystem

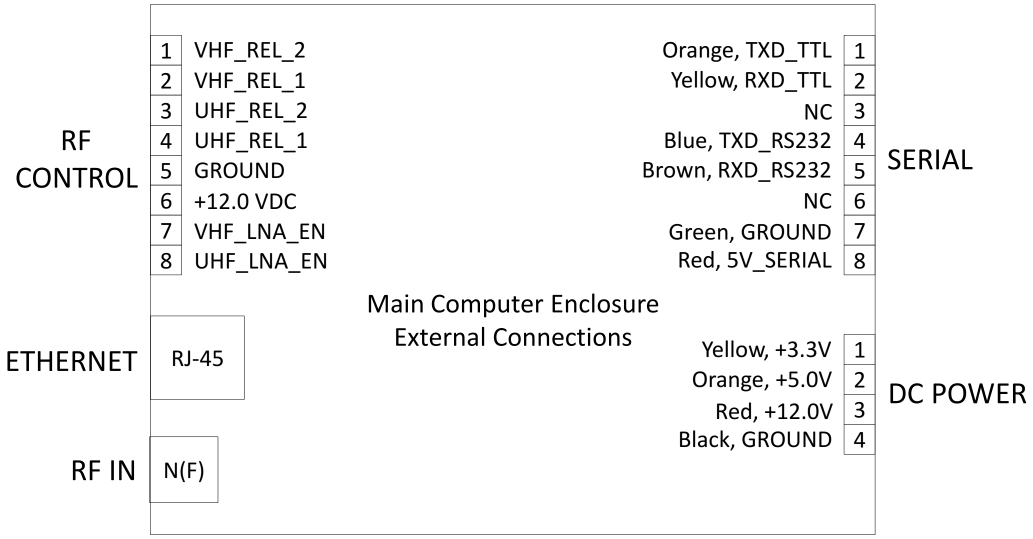


Figure 5.5: Main Computer Enclosure External Connections.

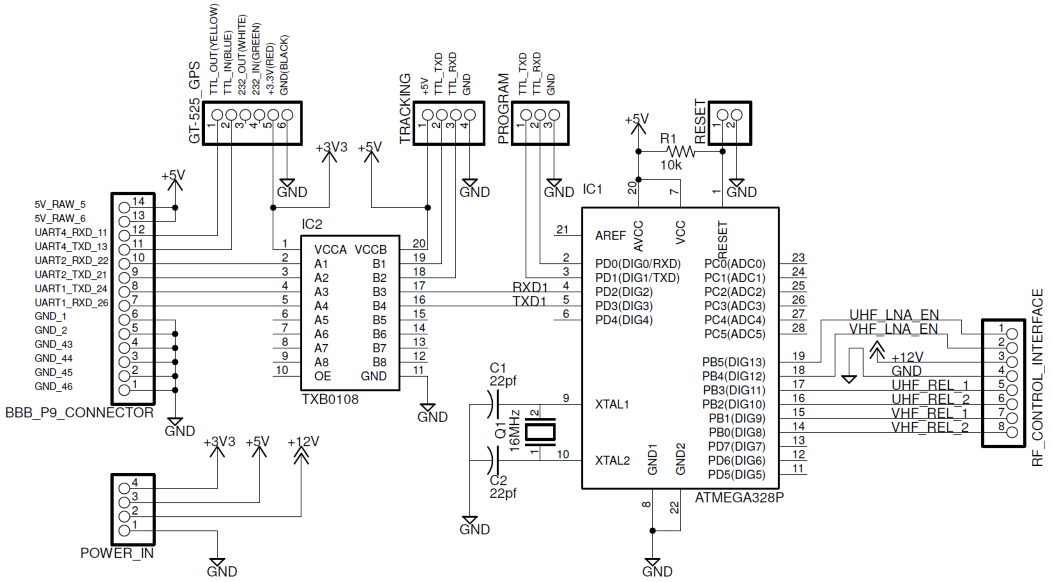


Figure 5.6: RF Control Microcontroller Schematic.

The Main Computer Enclosure contains the BeagleBone Black embedded computer, the FunCube Dongle Pro Plus SDR Receiver, and the RF Control Microcontroller Board. The BBB embedded computer communicates with the RF Microcontroller board and the Tracking Control Microcontroller board (both based on Arduinos) via UART. It also receives NMEA

0183 formatted GPS sentences via UART from a small GPS receiver. This enclosure is powered from the Power Supply Enclosure.

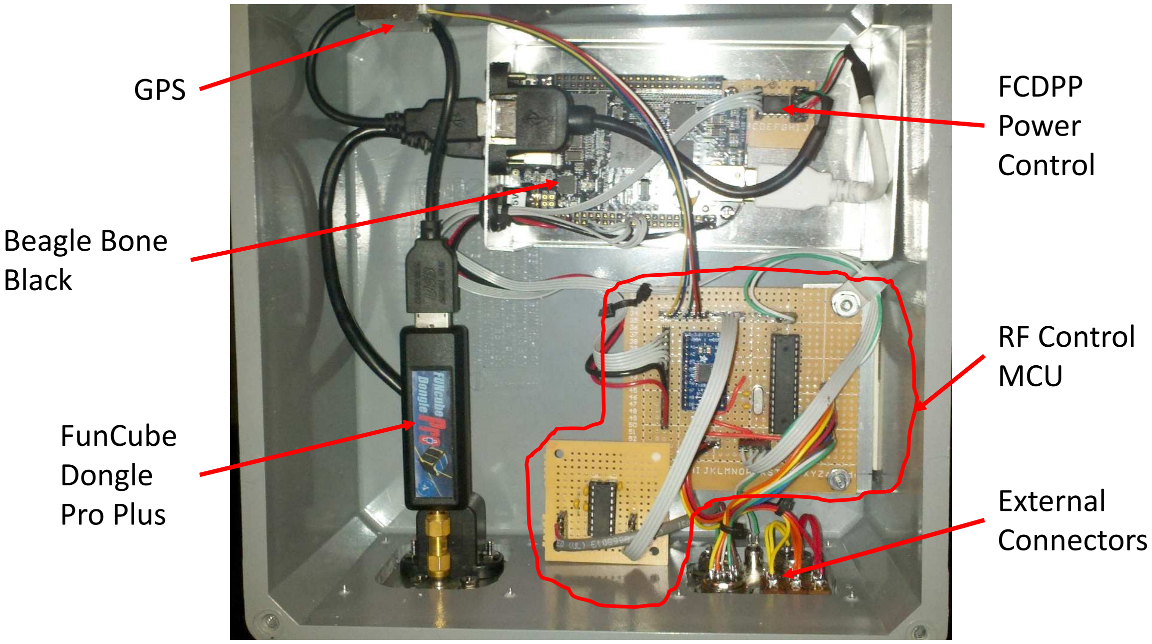
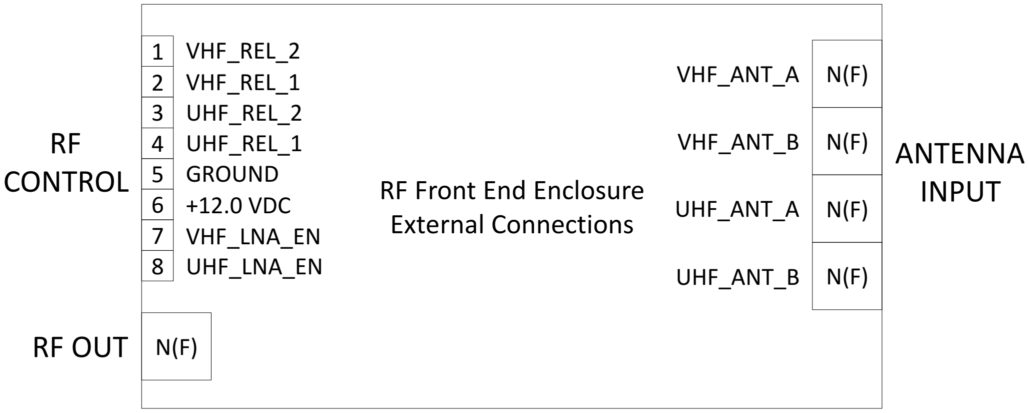


Figure 5.7: Main Computer Enclosure.

5.2.2 RF Frontend Subsystem



1	VHF_REL_2
2	VHF_REL_1
3	UHF_REL_2
4	UHF_REL_1
5	GROUND
6	+12.0 VDC
7	VHF_LNA_EN
8	UHF_LNA_EN

VHF_ANT_A	N(F)
VHF_ANT_B	N(F)
UHF_ANT_A	N(F)
UHF_ANT_B	N(F)

Figure 5.8: RF Front End Enclosure External Connections.

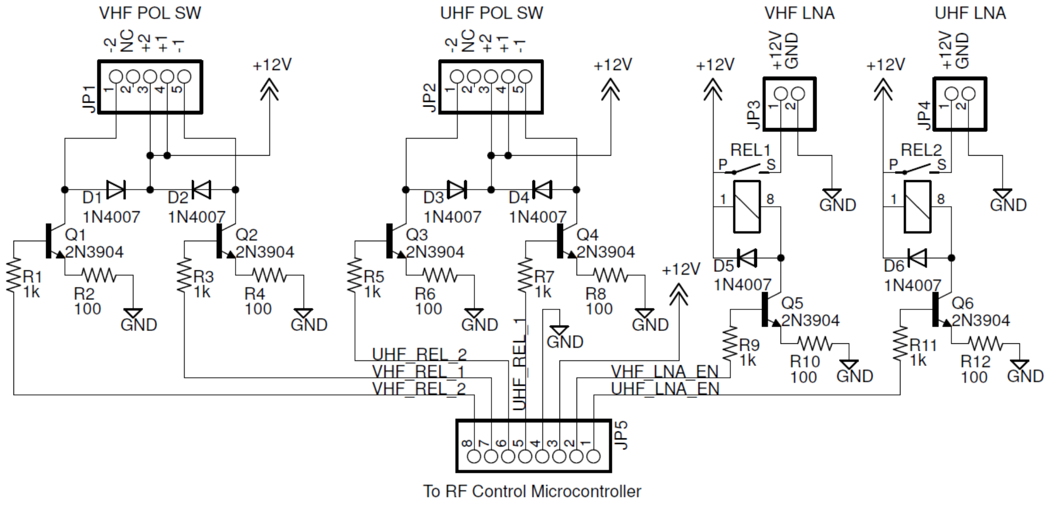


Figure 5.9: RF Control Interface Schematic.

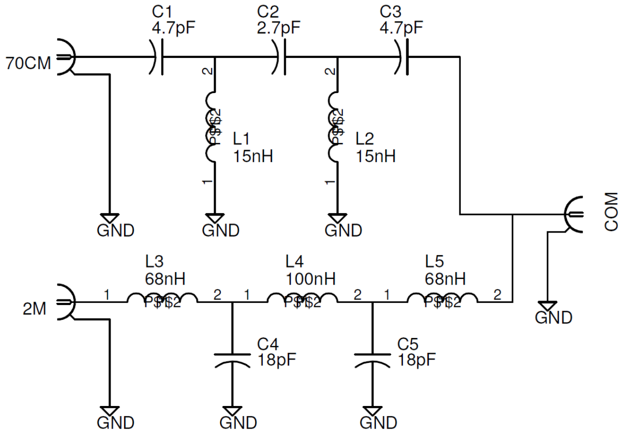


Figure 5.10: KI0AG Diplexer.

The RF Front End Enclosure contains the VHF polarization relay, the UHF polarization relay, the appropriate phasing lines for the polarization relays, the VHF Low Noise Amplifier (Advanced Receiver Research's P144VDG), the UHF Low Noise Amplifier (Advanced Receiver Research's P432VDG), the KI0AG 2m/70cm Diplexer, and the RF Control Interface Board. It should be noted that the KI0AG diplexer (also called the "arrow diplexer/duplexer") was fabricated from scratch utilizing information from [21].

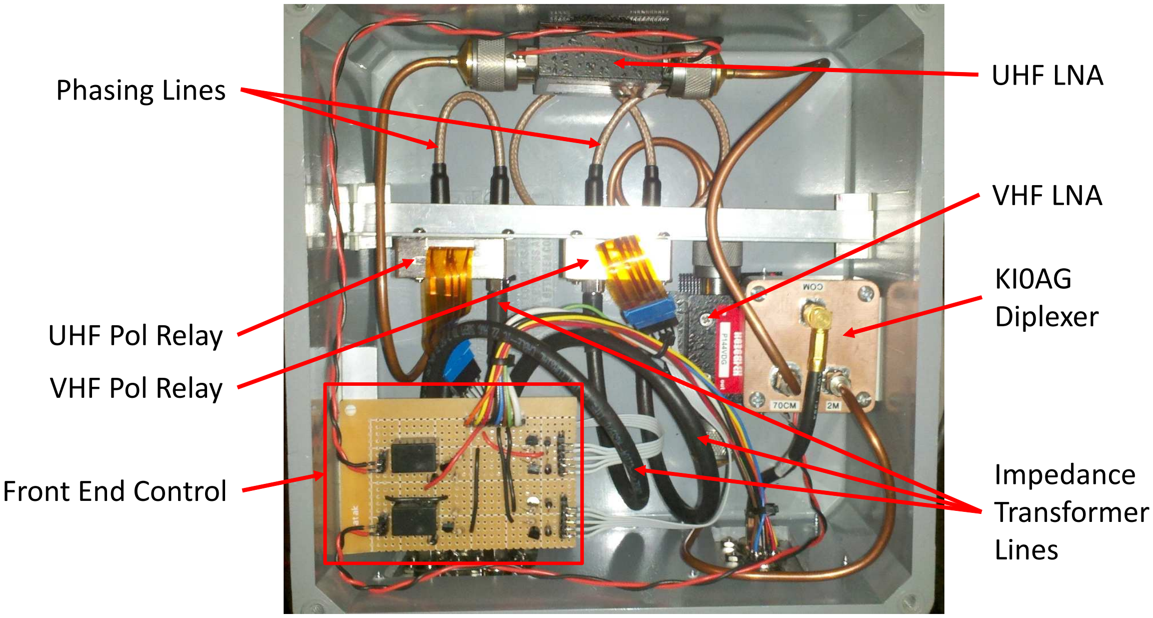


Figure 5.11: RF Front End Enclosure.

5.2.3 Antenna Subsystem

The antenna subsystem being utilized by the current prototype is a polarization sense selectable system. It is based on a design found in [22]. Figure 5.14 below shows how two linearly polarized Yagi antennas are connected, along with the appropriate phasing lines, impedance transformers, and a coaxial relay to produce circularly polarized radiation patterns. The polarization sense can be switched by changing the position of the coaxial relay. This is highlighted in Figure 5.12, where Right Hand Circular Polarization is shown. The antennas are oriented such that the pair from a respective band are at 90 degrees to each other along the long axis of the antennas with the two 2m antennas on the outer edge of the crossboom and the 70cm antennas towards the center (see [22] for a detailed description of the antenna configuration).

Circular polarization is important for communication with Low Earth Orbit cubesats. As mentioned previously, cubesats tend to use simple linear antennas. Because of the relatively simple stabilization systems, the orientation of the antennas is not always known absolutely. Thus as the antenna orientation changes throughout a pass the polarization at the receiving node changes. If linear antennas are used on the ground, the potential for misalignment between the antennas can result. This can result in deep signal loss, known as fades, between the spacecraft and receiving node. Potentially, if perfectly cross polarized, signal loss of over 30 dB can result. The use of circular polarization to receive a linearly polarized electromagnetic wave will result in a constant loss of 3 dB in the link budget. However, this loss is generally deemed an acceptable trade off compared to the potential for deep fades in the link.

Polarization of an electromagnetic field is defined by the orientation of the electric field. Circular polarization is defined by the direction of the rotation of the electric field using the right hand or left hand rule. The thumb is placed in the direction of propagation of the electromagnetic wave. If the electric field rotation follows the curling of the fingers of the right hand towards the palm then the polarization is said to be right hand circularly polarized (RHCP). If the electric field rotation follows the curling of the fingers of the left hand towards the palm, then the polarization is said to be left hand circularly polarized (LHCP).

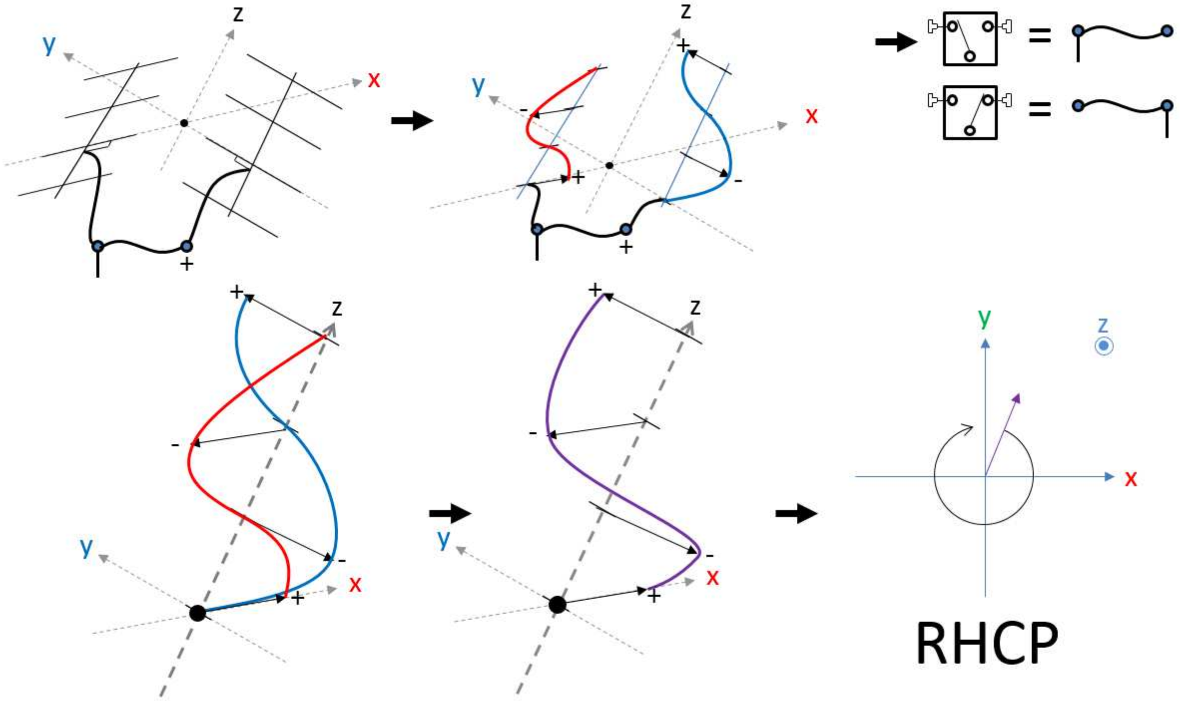
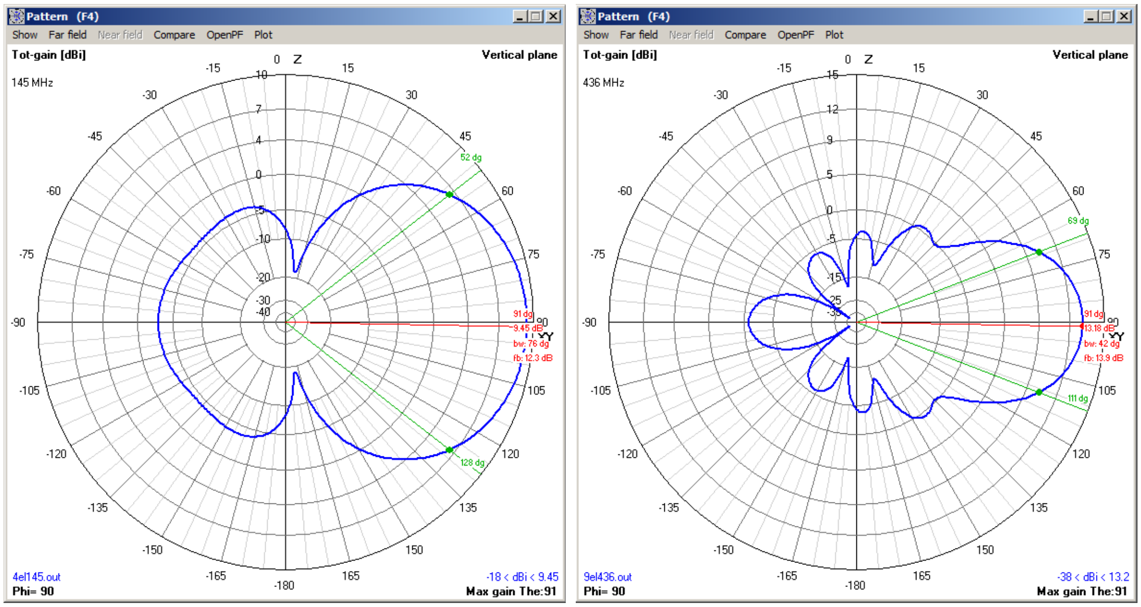


Figure 5.12: Combining Linear Yagis for Circular Polarization (RHCP).

The current 2m antennas are four element Yagis from Arrow Antennas, with a single antenna pattern shown in Figure 5.13a. The current 70cm antennas are 9 element home-made Yagis, with a single antenna pattern shown in Figure 5.13b. The antenna patterns were obtained using the 4NEC2 antenna modelling software. Based on the antenna simulations, the 2m antenna has a peak gain of 9.45 dBi and the 70cm antenna has a peak gain of 13.18 dBi.



(a) 4 Element VHF Arrow Antenna Pattern. (b) 9 Element UHF Antenna Pattern.

Figure 5.13: Prototype Yagi Antenna Patterns.

In addition to properly phasing the antennas, it is important to properly match the antenna impedances. Figure 5.14 shows the correct method of phasing these antennas. The antennas are tuned such that the nominal impedance is 50 Ω. Quarter wave transformers constructed out of 75 Ω coaxial cable transform the impedance to 100 Ω. The quarter wave transformers are connected to the coaxial relay using the method shown below. A 95 Ω quarter wave phasing line is used to create a 90 degree phase offset between the two antennas. The center conductor of the coaxial relay is presented with a 50 Ω impedance (two 100 Ω lines in parallel), resulting in a 50 Ω impedance.

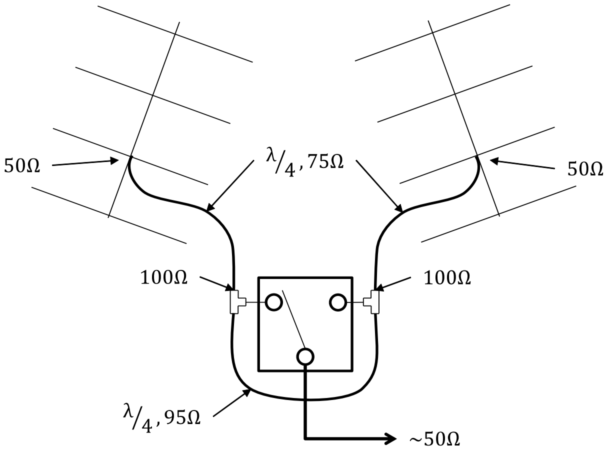


Figure 5.14: Antenna Circular Polarization Sense Control and Impedance Matching.

Also included in the antenna subsystem is the Pelco PT270-28P Tracking Pedestal and the postion feedback sensors. The feedback sensors utilize the I^{2}C protocol to communicate with the tracking control microcontroller board. For azimuth feedback the HMC5883L Triple Axis Magnetometer is utilized. For elevation feedback the MMA8452Q Triple Axis Accelerometer is utilized.

5.2.4 Tracking Control Subsystem

The Tracking Control Enclosure contains the Solid State Relay Bank Board, the Tracking Control Microcontroller Board, the azimuth 120VAC to 28VAC transformer, the elevation 120VAC to 28VAC transformer. It interfaces with the Pelco PT270-28P tracking pedestal, the magnetometer azimuth sensor (via I^{2}C), the accelerometer elevation sensor (via I^{2}C), and the Main Computer Enclosure (via UART). It is powered from the Power Supply Enclosure.

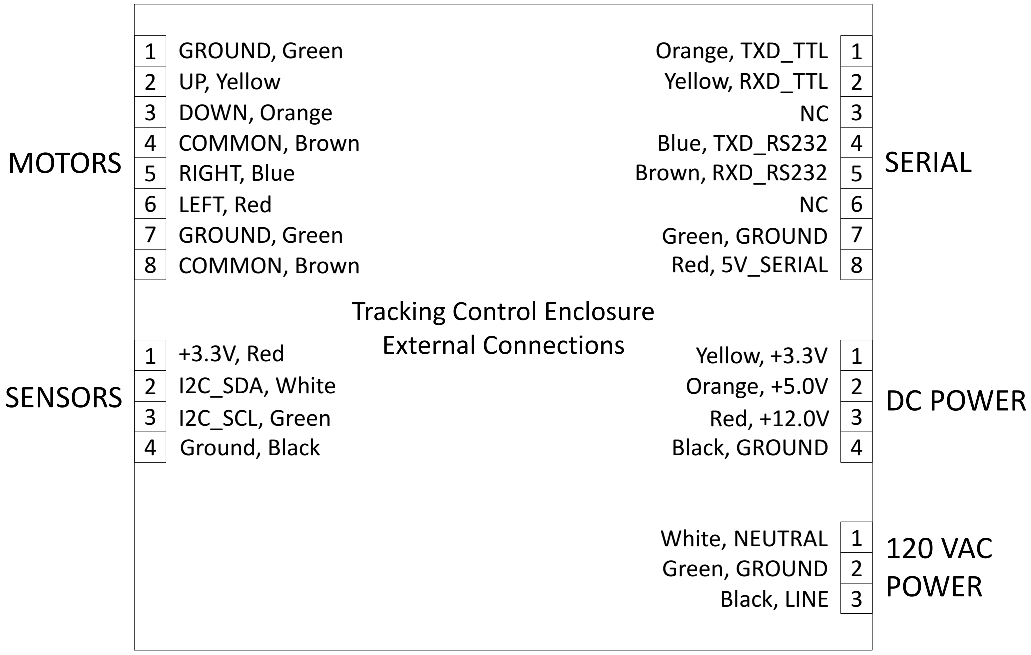


Figure 5.15: Tracking Control Enclosure External Connections.

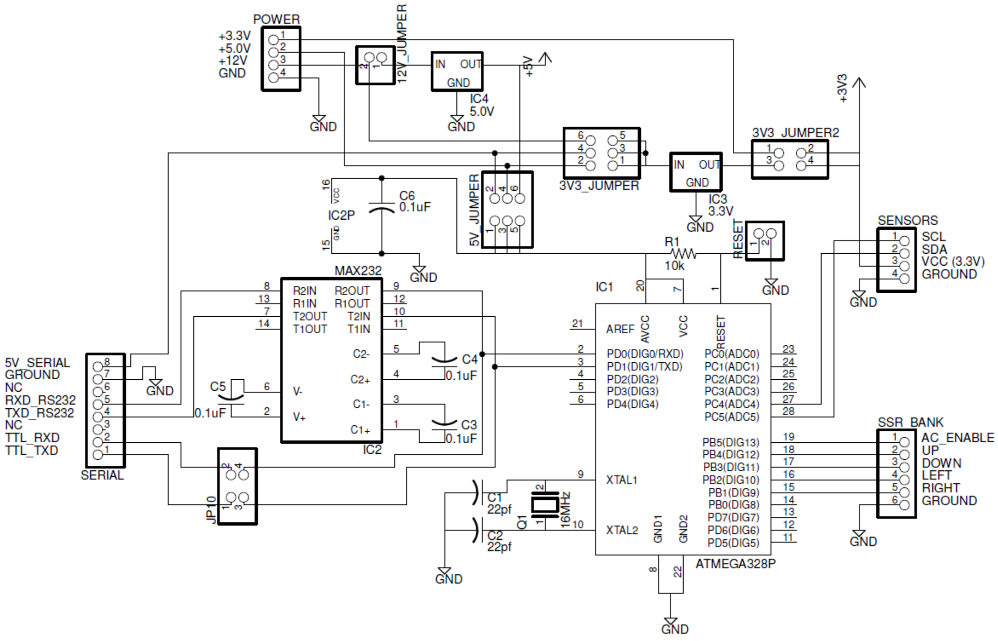


Figure 5.16: Tracking Control Microcontroller.

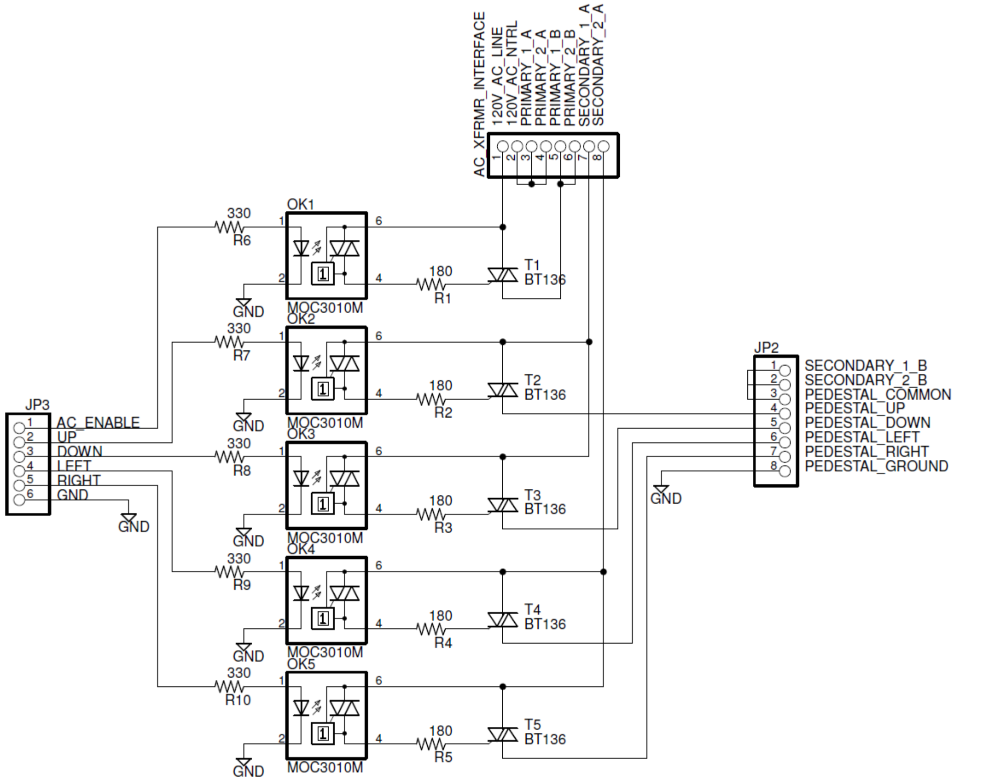


Figure 5.17: Solid State Relay Bank, Pelco PT270-28P Tracking Pedestal Interface.

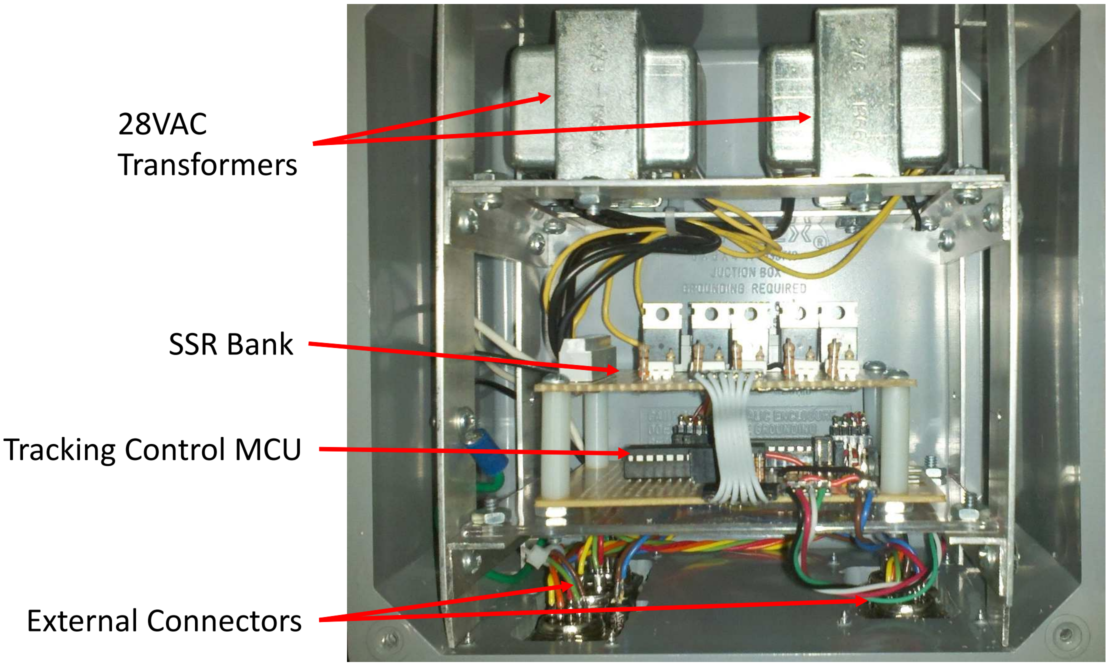


Figure 5.18: Tracking Control Enclosure.

5.2.5 Power Supply Subsystem

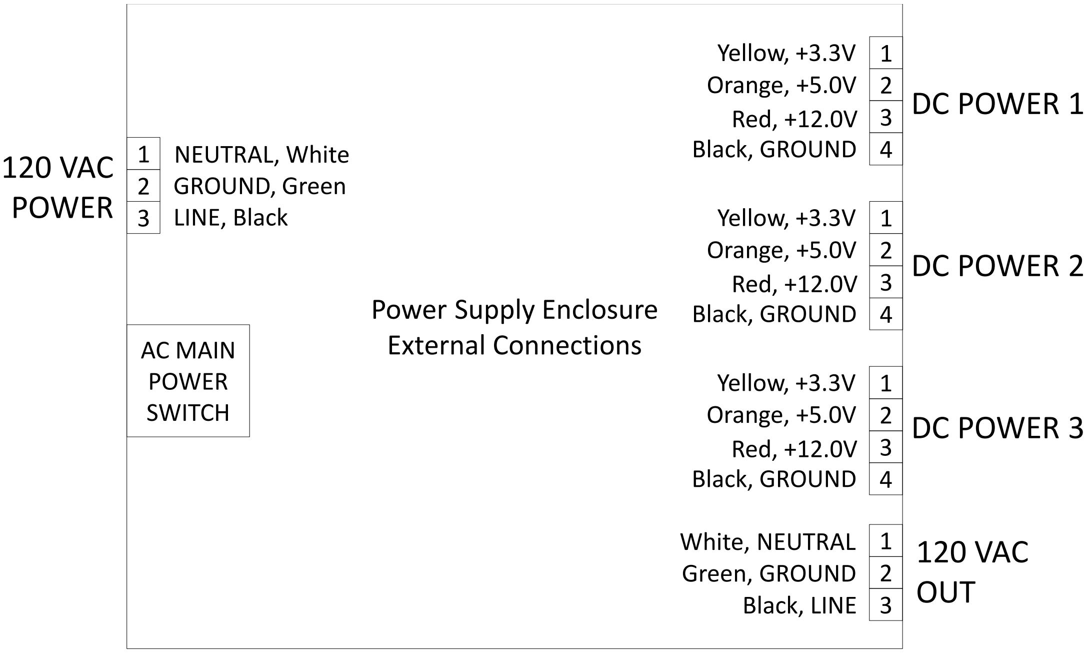


Figure 5.19: Power Supply Enclosure External Connections.

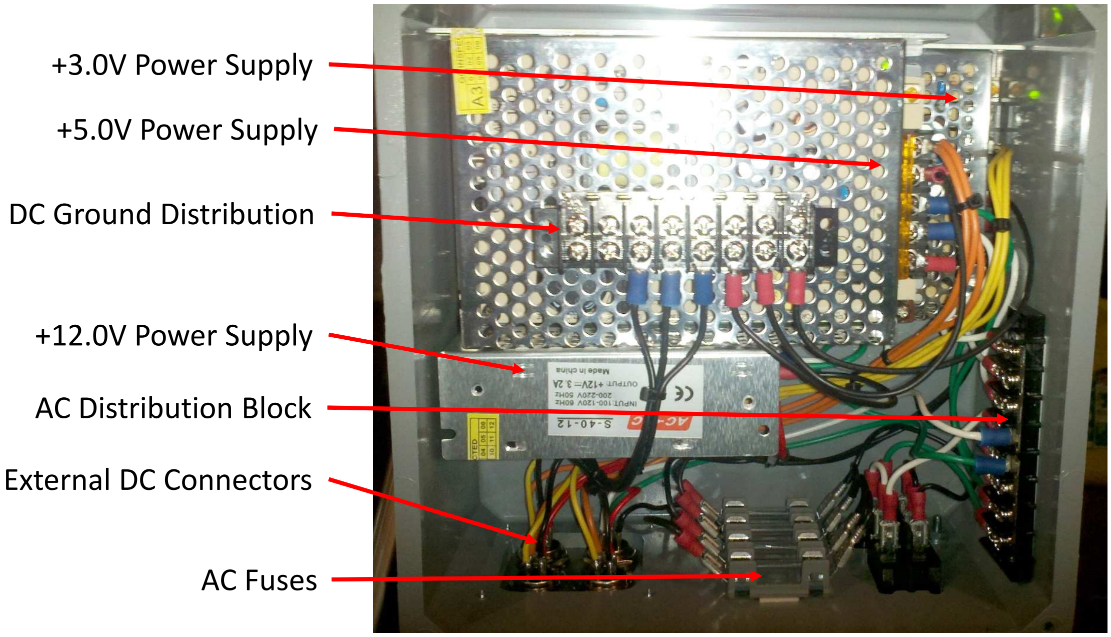


Figure 5.20: Power Supply Enclosure.

The Power Supply Enclosure contains three individual Linear DC power supplies, one for +3.3V, one for +5.0V, and one for +12.0V. It also contains individual AC fusing for each power supply as well as the external 120VAC power sent to the tracking control enclosure.

5.3 Receiver Node GNU-Radio Waveforms

Multiple GNU Radio waveforms, also called flowgraphs, have been written for this design for the various modulation schemes employed by existing cubesats. These flowgraphs were created using GNU-Radio Companion to generate a Python template. GNU Radio Companion (GRC) is a graphical interface for GNU Radio development that allows users to drag and drop signal processing blocks and then connect them to generate a waveform. GRC simply generates the Python code that glues the C/C++ signal processing blocks together. Further development is then done in the Python flowgraph in order to implement waveform specific features. These waveforms include Narrow Band FM (NBFM) demodulators, AFSK1200 decoders (using multimon), SSB/CW demodulators, and direct sample capture flowgraphs. This section will go over each waveform in more detail. All of these flowgraphs were heavily based on the examples provided in [23].

In all cases the connection to the FCDPP is the same. As described in Section 5.1.1, the FCDPP receives 192 kHz of RF spectrum and passes IQ data to the host. In all of the flowgraphs written so far, the samples are then converted into the GNU Radio Complex data type using the float to complex block.

There is a strong DC component at the tuned center frequency as shown in Figure 5.21 below. 144.39 MHz is the example center frequency for the following explanation. In order to avoid this DC component, the FCDPP is tuned exactly 48 kHz *lower* than the desired center frequency of a spacecraft's downlink, as depicted in Figure 5.22 below. The GNU Radio Source block, as described in more detail later, then retunes the complex data stream by +48 kHz to re-center the downlink frequency. Finally, the data stream, at a rate of 192 ksps is decimated by 4 to produce a complex data stream of 48 ksps. Again since this includes both I and Q data, the data stream is 48 kHz wide. This bandwidth is wide enough to contain the entire downlink signal even accounting for Doppler shift (±12 kHz at UHF frequencies) and a signal that is 20 kHz wide.

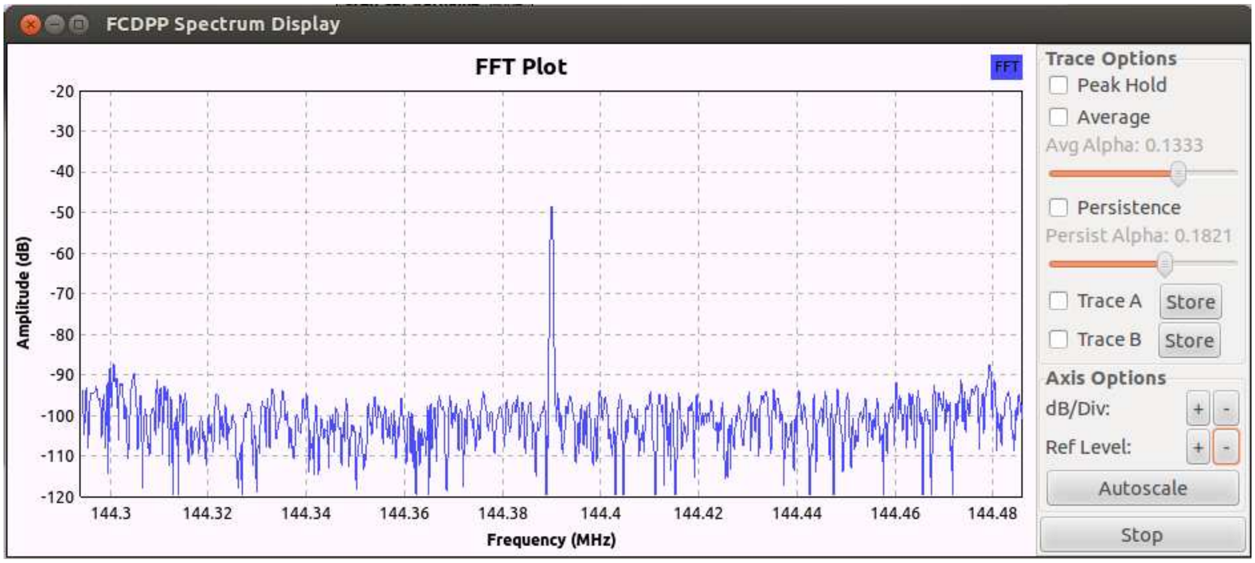
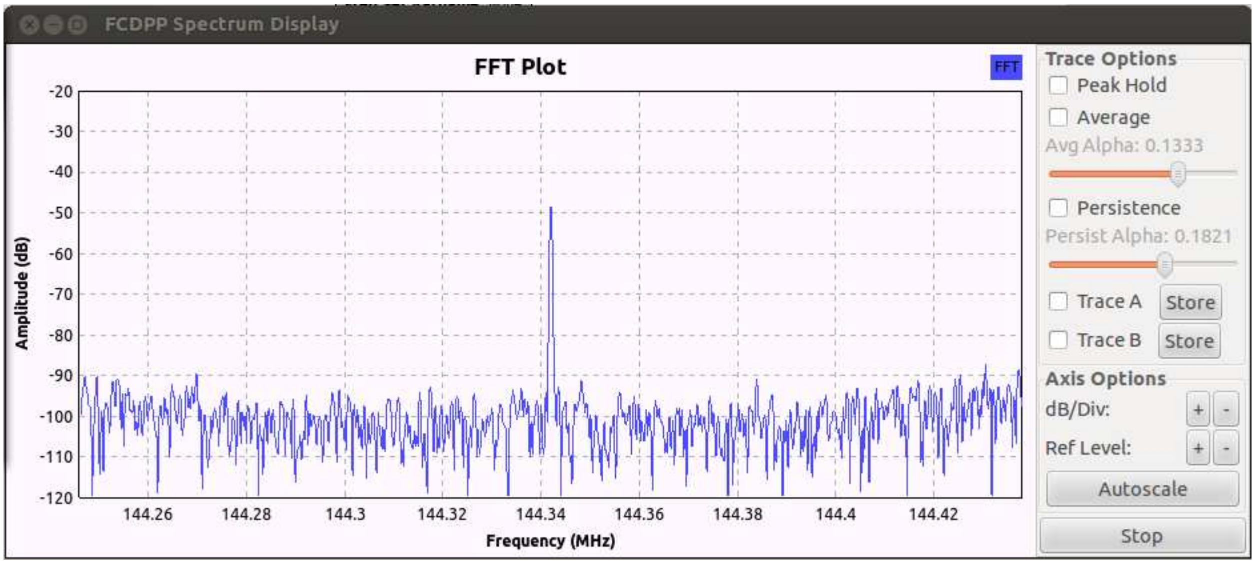
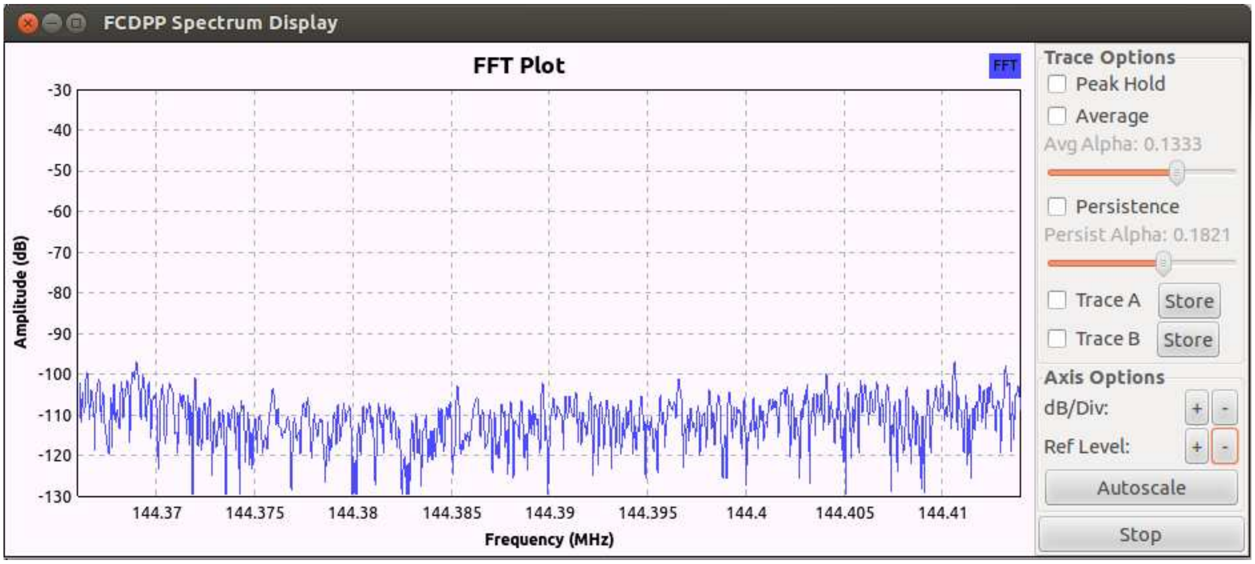


Figure 5.21: FCDPP Strong DC component GNU-Radio Hierarchical Source Block.



□ Peak Hold
□ Average
Avg Alpha: 0.1333
□ Persistence
Persist Alpha: 0.1821
□ Trace A
□ Trace B

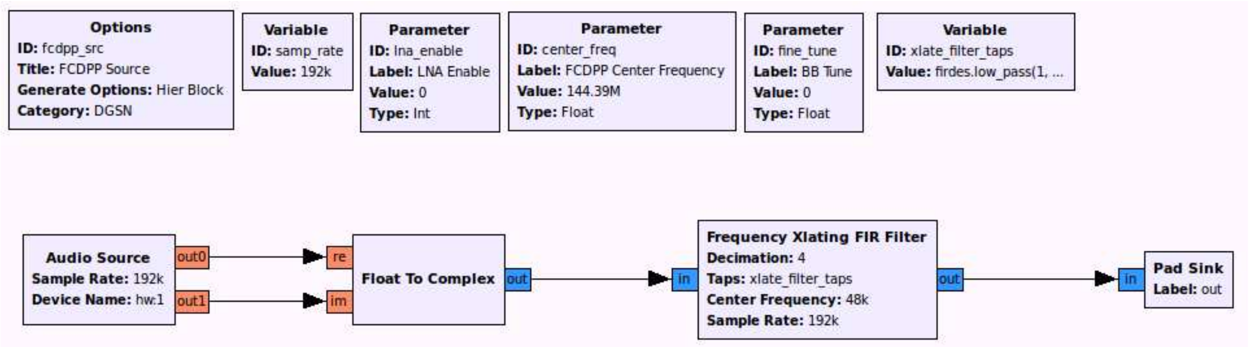
Figure 5.22: FCDPP Tuned 48 kHz lower than the desired center frequency.



□ Peak Hold
□ Average
Avg Alpha: 0.1333
□ Persistence
Persist Alpha: 0.1821
□ Trace A
□ Trace B

Figure 5.23: Frequency Shift by +48kHz and Decimation by a Factor of 4.

As described in Section 5.1.3, access to the FCDPP Source block was created as a Hierarchical Block in GNU-Radio. This FCDPP source block is utilized by all of the flowgraphs and accomplishes a number of required signal processing tasks. First, access to the FCDPP is accomplished through the use of the soundcard source block. Then the parallel streams of Real and Complex Data, at a rate of 192 kilo-samples per second, are converted to a Complex Data type. Then the Frequency Translating FIR filter block to accomplish a number of tasks, including adjusting the frequency back by +48 kHz (to avoid the unwanted DC component), decimating by a factor of 4 (to reduce the sample rate to 48 kHz), filtering (to remove images from the decimation process). Finally an output pad is provided for connection to further signal processing blocks downstream. The combined effect of this block, when used in other waveforms, is that it handles all of the necessary interfacing with the FCDPP. This block tunes the radio to the desired frequency (48 kHz less than the desired true frequency), controls powering of the internal FCDPP LNA, and presents a 48 kHz wide stream of samples at the output for further processing. It also conducts logging operations, storing timestamped information concerning the operation of the FCDPP (frequency tuning events, LNA power on/off events, etc.). The template for this block was generated in GNU-Radio Companion and is shown in Figure 5.24. Further development in the Python code was then completed in order to implement a number of automatic features (such as sending commands to tune the FCDPP and control the internal LNA power via the `fcdproplus-control` utility).



Options
ID: fcdpp_src
Title: FCDPP Source
Generate Options: Hier Block
Category: DGSN

Variable
ID: samp_rate
Value: 192k

Parameter
ID: lna_enable
Label: LNA Enable
Value: 0
Type: Int

Parameter
ID: center_freq
Label: FCDPP Center Frequency
Value: 144.39M
Type: Float

Parameter
ID: fine_tune
Label: BB Tune
Value: 0
Type: Float

Variable
ID: xlate_filter_taps
Value: firdes.low_pass(1, ...

Audio Source
Sample Rate: 192k
Device Name: hw:1

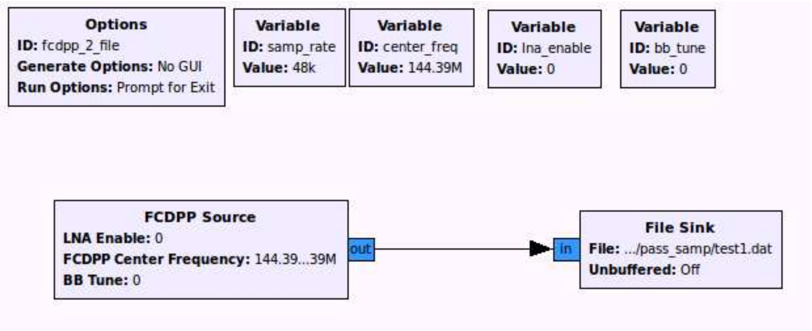
Float To Complex

Frequency Xlating FIR Filter
Decimation: 4
Taps: xlate_filter_taps
Center Frequency: 48k
Sample Rate: 192k

Pad Sink
Label: out

Figure 5.24: FunCube Dongle Pro Plus Hierarchical Source Block Flowgraph.

The simplest flowgraph utilized by the receiver node is the direct sample capture flowgraph. This waveform simply records the raw samples output from the GNU Radio source block described above. Again a template for this waveform was generated using GRC and then additional features were implemented in the Python code for the flowgraph, such as a command line interface to control the FCDPP (through the hierarchical block described above), control over start time, stop time, and duration of the recording, as well as the implementation of logging features. The GRC flowgraph is shown in Figure 5.25. This flowgraph generates GNU Radio Complex Data Type files and is recorded at a rate of 48 kilo-samples per second, thus generates recordings of 48 kHz of RF spectrum for the duration specified in the CLI. This waveform allows spacecraft to be tracked for which a flowgraph that actually demodulates the signals from the spacecraft has yet to be written.



Options
ID: fcdpp_2_file
Generate Options: No GUI
Run Options: Prompt for Exit

Variable
ID: samp_rate
Value: 48k

Variable
ID: center_freq
Value: 144.39M

Variable
ID: lna_enable
Value: 0

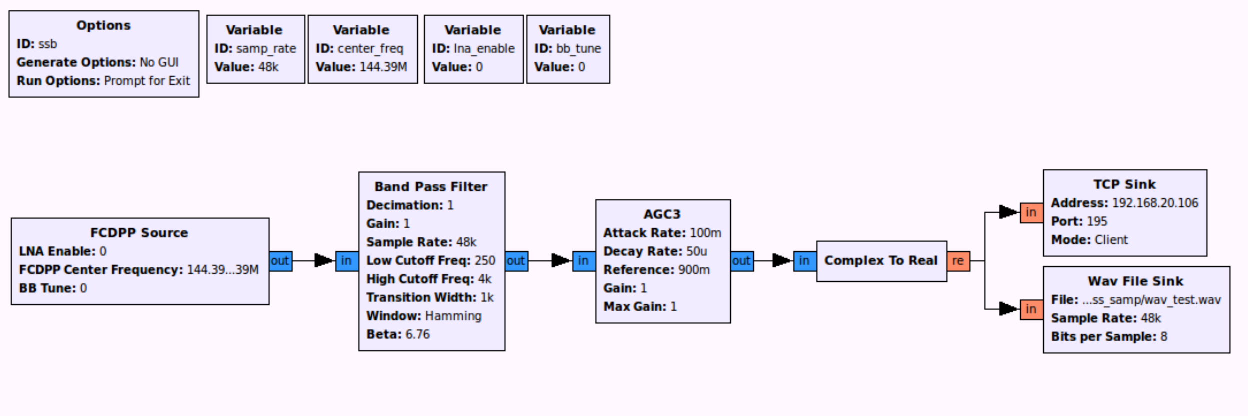
Variable
ID: bb_tune
Value: 0

FCDPP Source
LNA Enable: 0
FCDPP Center Frequency: 144.39...39M
BB Tune: 0

File Sink
File: .../pass_samp/test1.dat
Unbuffered: Off

Figure 5.25: Direct Sample Capture Flowgraph.

Another potentially useful flowgraph is a Single Side Band (SSB) or CW (a.k.a. Morse Code) demodulator. This flowgraph is depicted in Figure 5.26 below. Essentially this flowgraph, utilizing the FCDPP source block, simply has a channel filter, automatic gain control, and then an envelope detector. The output of the flowgraph is either a standard .wav file recording (at 48 kHz) of the audio, or a potential TCP sink in order to stream the audio to a remote computer. Further work can be done with this flowgraph to include an actual Morse Code decoder to print the decoded information to the terminal window or a file.



Options
ID: ssb
Generate Options: No GUI
Run Options: Prompt for Exit

Variable
ID: samp_rate
Value: 48k

Variable
ID: center_freq
Value: 144.39M

Variable
ID: lna_enable
Value: 0

Variable
ID: bb_tune
Value: 0

FCDPP Source
LNA Enable: 0
FCDPP Center Frequency: 144.39...39M
BB Tune: 0

Band Pass Filter
Decimation: 1
Gain: 1
Sample Rate: 48k
Low Cutoff Freq: 250
High Cutoff Freq: 4k
Transition Width: 1k
Window: Hamming
Beta: 6.76

AGC3
Attack Rate: 100m
Decay Rate: 50u
Reference: 900m
Gain: 1
Max Gain: 1

Complex To Real

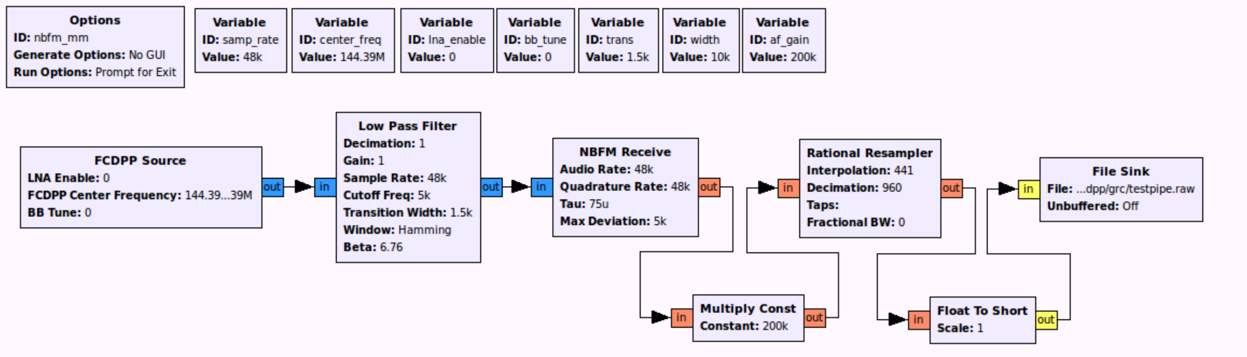
TCP Sink
Address: 192.168.20.106
Port: 195
Mode: Client

Wav File Sink
File: ...ss_samp/wav_test.wav
Sample Rate: 48k
Bits per Sample: 8

Figure 5.26: Single Side Band and CW Demodulator.

As AFSK1200 using the AX.25 UI frame format is a very common modulation scheme, a flowgraph to support this was also developed. Figure 5.27 below shows the GRC flowgraph for the Narrow Band FM Demodulator that is required for this modulation scheme. It is important to note that the figure only portrays a portion of the total waveform. Essentially, the FCDPP source, channel filter, and NBFM Receive block comprise the FM receiver. At this point in the flowgraph we essentially have demodulated audio, at a rate of 48 kHz. This is functionally equivalent to the audio output port of a traditional hardware FM radio. In order to actually demodulate the audio tones of the AFSK modulation scheme and then decode the AX.25 UI frame, more processing is required. This is accomplished through the use of a utility called *multimon*, which is a decoding program written by Tom Sailer for modulation schemes common to Amateur Radio operations, which supports AFSK1200/AX.25 decoding.

In order to "connect" the audio output of the flowgraph to the audio input of *multimon* a Linux FIFO pipe is utilized. All blocks following the NBFM Receive block are utilized to condition the audio to the correct sample rate and levels expected by the *multimon* utility. The properly conditioned audio is passed through the Linux pipe to the input of the decoder. When the proper sequence of tones is demodulated by the NBFM receiver and passes through the pipe, *multimon* demodulates the audio tones, and then decodes the AX.25 packet. As far as the multimon utility is concerned, it thinks it is reading a file in "raw" format that happens to be infinite in length. The net effect of this combination is that a realtime AFSK1200 decoder is running and as packets are received from the spacecraft they are printed to either the terminal window or dumped to a file. The automatic launching of multimon is accomplished through code modifications to the Python waveform generated by GRC. This waveform has been tested extensively using the Amateur Radio Automatic Packet Reporting System (APRS) that also utilizes AFSK1200 modulation scheme and can be found locally on frequency 144.39 MHz.



Options
ID: nbfm_mm
Generate Options: No GUI
Run Options: Prompt for Exit

Variable
ID: samp_rate
Value: 48k

Variable
ID: center_freq
Value: 144.39M

Variable
ID: lna_enable
Value: 0

Variable
ID: bb_tune
Value: 0

Variable
ID: trans
Value: 1.5k

Variable
ID: width
Value: 10k

Variable
ID: af_gain
Value: 200k

FCDPP Source
LNA Enable: 0
FCDPP Center Frequency: 144.39...39M
BB Tune: 0

Low Pass Filter
Decimation: 1
Gain: 1
Sample Rate: 48k
Cutoff Freq: 5k
Transition Width: 1.5k
Window: Hamming
Beta: 6.76

NBFM Receive
Audio Rate: 48k
Quadrature Rate: 48k
Tau: 75u
Max Deviation: 5k

Multiply Const
Constant: 200k

Rational Resampler
Interpolation: 441
Decimation: 960
Taps:
Fractional BW: 0

Float To Short
Scale: 1

File Sink
File: ...dpp/grc/testpipe.raw
Unbuffered: Off

Figure 5.27: Narrow Band FM Demodulator with linux FIFO pipe output.

Another important note is that *multimon* not only supports AFSK1200, but also FSK9600. The same flowgraph above, with simple parameter modification (such as the bandwidth of the channel filter) could in theory be used to decode the G3RUH FSK modulation scheme that is also very common in cubesat communications. However, this has not been tested as of the writing of this document, and therefore the feasibility of this cannot be commented on.

Chapter 6

Link Budget Analyses & Extended Network Simulations

This section will present a number of link budget analyses. A number of assumptions must be made, but the goal is to provide a fair comparison between various scenarios. This section will tie together the information presented in Chapters 3, 4, & 5 to work through various representative scenarios. Further simulations and the associated results are presented to show the benefits of switching to digital modulation schemes with forward error correction, the vastly increased access time that a ground station network can support, and the utility of the candidate receiver node design.

6.1 Scenario Parameters & Constraints

The SwissCube orbit will be chosen for the simulation data presented. It is important to note that the link budget analyses presented are *not* specific to the actual SwissCube spacecraft, but rather use the representative orbit of the spacecraft for the analysis. This particular orbit is chosen because it has neither the highest nor lowest orbital altitude, thus making it somewhat representative of the various spacecraft.

Three modulation schemes will be presented in the following analyses. First AFSK, which is commonly associated with the 2m frequencies will be investigated. Then G3RUH FSK which is commonly associated with 70cm frequencies will be investigated. Finally, BPSK with Convolutional Coding FEC is presented. Since the IARU is no longer authorizing downlinks in the 2m Amateur Satellite Sub-band for spacecraft that do not provide a voice or data relay service to the Amateur Radio community, only the 70cm band will be analyzed for the BPSK modulation scheme scenario. Table 6.1 below lists the parameters & constraints that are common to each scenario presented.

Table 6.1: Link Budget Simulation Parameters & Constraints.

Parameter	**Value**	**Note**
Spacecraft Avg Orbit Altitude	725 [km]	SwissCube Type Orbit
Spacecraft Inclination Angle	98 [deg]	SwissCube Type Orbit
Spacecraft Transmit Power	0.0 [dBW]	1 Watt, assumption
Spacecraft Antenna Gain	2.0 [dBi]	Simple Dipole Pattern
Spacecraft Antenna Polarization	Linear	Simple Dipole Polarization
Polarization Losses	3.0 [dB]	Linear to Circular Pol Loss
Channel Bandwidth	20 [kHz]	from Section 2.3
Atmospheric Losses (2m)	0.2 [dB]	approximation
Ionospheric Losses (2m)	0.7 [dB]	approximation
Atmospheric Losses (70cm)	0.35 [dB]	approximation
Ionospheric Losses (70cm)	0.4 [dB]	approximation
Implementation Loss	1.0 [dB]	
RX Component Noise Temp (2m)	69.463 [K]	from Section 5.1.4
RX Component Noise Temp (70cm)	94.020 [K]	from Section 5.1.4
RX Antenna Temperature	290 [K]	Earth Noise Temperature
Target BER to Close Link	10^{-5}	Section 3.2

6.2 Link Budget Equations

6.2.1 Variable Description

When performing link budget analyses there are a multitude of factors that must be taken into account, and often a number of assumptions must be made. Due to this, there are a large number of variables used to perform the necessary calculations. Various sources may use different nomenclature and variables to describe the mathematical quantities. Therefore, this section will list out the variables used for the analysis with a short description of each variable along with the units utilized.

Table 6.2: Link Budget Variable Description.

Parameter	Variable	Unit	Description
Link Frequency	f_0	MHz	Frequency of operation
Link Wavelength	λ_0	m	Wavelength of operation
Speed of Light	c	m/s	constant, value: 299,792,458
Required CNR	CNR_{req}	dB	Theoretical CNR required for target BER
Receiver Implementation Loss	L_{impl}	dB	Theoretical to Real Receiver
Required Effective CNR	CNR_{req_eff}	dB	Actual CNR required for target BER
Spacecraft Transmit Power	P_{tx}	dBW	1 Watt, assumption
Spacecraft Antenna Gain	G_{tx_ant}	dBi	Simple Dipole Pattern
Spacecraft EIRP	$EIRP_{tx}$	dBW	S/C Effective Isotropically Radiated Power
Polarization Losses	L_{pol}	dB	Linear to Circular Polarization Loss
Atmospheric Losses	L_{atmo}	dB	Atmospheric Absorption Loss
Ionospheric Losses	L_{iono}	dB	Ionospheric Loss
Slant Range	d	km	Distance between S/C and ground node
Path Loss	L_{path}	dB	Path Loss
Path Loss Exponent	n		Path Loss Exponent
Free Space Path Loss	L_{fs}	dB	Free Space Path Loss
Receive Antenna Gain	G_{rx_ant}	dBi	Receive Antenna Gain
Receive Signal Power	P_r	dBW	Receive Signal Power
RX Component Noise Temperature	T_{comp}	K	Noise temperature of receive components
RX Antenna Temperature	T_{ant}	K	Receive Antenna Noise Temperature
RX System Noise Temperature	T_{sys}	K	Receive System Noise Temperature
Boltzmann's Constant	k	dBW/K/Hz	constant, value: -228.6
Receive Bandwidth	B	Hz	Receiver Noise Bandwidth
Receive Noise Power	P_n	dBW	Receiver Noise Power
Link Margin	M_{link}	dB	Link Margin

6.2.2 Link Equations

This section will describe the specific equations and associated variables used to perform the link budget analyses.

Transmit

The power radiated by the spacecraft is the proper combination of the radio's output power as well as the gain of the antenna system utilized by the spacecraft. This quantity is known as the Effective Isotropically Radiated Power (EIRP). For the following analyses it is assumed that the power output of the radio is 1.0 W, or 0.0 dBW in the logarithmic scale. It is also assumed that the spacecraft has an omni-directional antenna pattern achieving gains that are close to that of a simple dipole. For the analyses presented, it is assumed that the antenna gain is 2.0 dBi. Equation 6.1 below shows the equation for calculating the EIRP of the spacecraft in the logarithmic (dB) scale.

$$EIRP_{tx} = P_{tx} + G_{tx\_ant}\ [dBW] \tag{6.1}$$

Path Loss

One of the most important parameters that must be determined for an accurate link budget analysis is the Path Loss between the spacecraft and a given ground station node. This value is determined based off of two key parameters, the wavelength of operation and the link distance, or Slant Range. The slant range is the line of site distance between the spacecraft and the ground station node. For this analysis, the assumption is made that the path between the spacecraft and the receiver node is unobstructed and thus the path loss is the Free Space Path Loss, with path loss exponent of n = 2.

$$\lambda_0 = \frac{c}{f_0}\,[m] \tag{6.2}$$

$$L_{path} = 10\log_{10}\left(\frac{4\pi d}{\lambda_0}\right)^n\,[dB] \tag{6.3}$$

$$L_{fs} = 10\log_{10}\left(\frac{4\pi d}{\lambda_0}\right)^2\,[dB] \tag{6.4}$$

Received Power

The power received at the ground station node must account for multiple factors. This value is used to determine the carrier to noise ratio (CNR). The received power is calculated by subtracting all losses from the EIRP of the spacecraft. Not only is the Free Space Path Loss a factor, but atmospheric and ionospheric factors must also be accounted for. Additionally, for the analyses presented, it is assumed that the spacecraft is utilizing a linearly polarized antenna system while the ground station receiving node is utilizing a circularly polarized antenna. This polarization mismatch results in a 3 dB loss, which must also be accounted for. Finally, the receive antenna gain must also be accounted for as it increases the power received by the ground station node. The equation to calculate the received power, in the logarithmic scale, is given in Equation 6.5 below.

$$P_r = EIRP_{tx} - L_{atmo} - L_{iono} - L_{pol} - L_{fs} + G_{rx\_ant}\,[dBW] \tag{6.5}$$

Receiver Noise

Now that a method for calculating the received power has been established, a method is needed for calculating the receiver noise power. These two quantities, the received power and the receiver noise power, then allow for the Carrier to Noise Ratio to be calculated, which leads to a determination of bit error rates. Equation 6.6 below shows the calculation for Noise Power in the logarithmic (dB) scale. the variable k is known as Boltzmann's constant, and has a value of -228.6 dBW/Hz/K. The variable T is the noise temperature of the system, which accounts for the noise generated by both active and passive components of the receive chain as well as the temperature of the antenna system. The value of T for both the 2m and 70cm receive chains of the prototype system is presented in Section 5.1.4. Finally, the variable B is the noise bandwidth of the receiving system.

$$P_n \;=\; k \,+\, T \,+\, B\,[dB] \tag{6.6}$$

Carrier to Noise Ratio

The overall goal of the link budget calculation is to determine a Carrier to Noise Ratio (CNR) at the receiving system. The CNR of the link can be calculated using Equation 6.7. This received CNR is then compared against the CNR that is required by a given modulation scheme to achieve the target Bit Error Rate (BER). Another item that must be accounted for is implementation loss. Implementation loss is a way of accounting for the differences between ideal or theoretical BER curves and the actual values that can be expected from a real world implementation. This implementation loss then increases the required CNR to a value referred to as the effective required CNR, as shown by Equation 6.8. Finally, The link margin can be determined. The link margin represents how much excess carrier to noise ratio exists. When the link margin is positive the link is said to be "closed" and the BER of the link will be below the required BER. If the link margin is negative then the link is not closed and the BER of the link will be above the required BER of the link. Link Margin is calculated according to Equation 6.9.

$$CNR \;=\; P_r - P_n\,[dB] \tag{6.7}$$

$$CNR_{req\_eff} \;=\; CNR_{req} + L_{impl}\,[dB] \tag{6.8}$$

$$M_{link} \;=\; CNR - CNR_{req\_eff}\,[dB] \tag{6.9}$$

6.3 AFSK_FM & G3RUH_FSK Link Budget Analyses

In this section link budget analyses are conducted for the AFSK_FM and G3RUH_FSK modulation schemes. Again, the representative orbit is the same as the SwissCube Orbit. In both cases the 10 degree elevation constraint is still in place. The link distance used for the analysis is the distance between the satellite and each ground station node, averaged over every AOS and LOS distance to all 19 nodes throughout the simulation scenario of one month, when it is at 10 degrees elevation.

As mentioned previously, for the AFSK_FM Analysis, the frequency of operation is assumed to be the center of the 2m Amateur Satellite Service sub-band, specifically 145.900 MHz. Similarly, for the G3RUH_FSK analysis, the frequency of operation is assumed to be the center of the 70cm Amateur Satellite Service sub-band, specifically 436.500 MHz. In both cases, it is assumed that the prototype receiver node design is utilized for the link analysis. Specifically, what this means is that it is assumed that the directional yagi antennas, along with the tracking pedestal capabilities are being utilized. For the noise performance of the system, the noise calculations presented in Section 5.1.4 are used.

6.3.1 AFSK_FM Link Budget

The results of the link budget analysis are presented in Table 6.3 below. As shown in the final line, the link margin is +0.81 dB. This means that the link does close for the target BER of 10^{-5}. To re-iterate, this analysis is only valid for the moment in time when the satellite is at an elevation of 10 degrees.

Table 6.3: AFSK_FM Link Budget Analysis.

Parameter	**Variable**	**Value**	**Unit**	**Note**
Link Frequency	f_0	145.900	MHz	Center of sub-band
Link Wavelength	λ_0	2.054780	m	used for path loss calculation
Link Modulation				AFSK_FM
Required CNR	CNR_{req}	23.2	dB	for 10^{-5} BER, see Fig. 3.2
Receiver Implementation Loss	L_{impl}	1.0	dB	
Required Effective CNR	CNR_{req_eff}	24.2	dB	
Spacecraft Transmit Power	P_{tx}	0.0	dBW	1 Watt, assumption
Spacecraft Antenna Gain	G_{tx_ant}	2.0	dBi	Simple Dipole Pattern
Spacecraft EIRP	$EIRP_{tx}$	2.0	dBW	
Polarization Losses	L_{pol}	3.0	dB	Linear to Circular Pol Loss
Atmospheric Losses (2m)	L_{atmo}	0.2	dB	approximation
Ionospheric Losses (2m)	L_{iono}	0.7	dB	approximation
Slant Range	d	2198.339	km	Refer to text
Path Loss	L_{fs}	142.57	dB	
Receive Antenna Gain	G_{rx_ant}	9.45	dBi	from Section 5.2.3
Receive Signal Power	P_r	-135.02	dBW	
RX Component Noise Temp (2m)	T_{comp}	69.463	K	from Section 5.1.4
RX Antenna Temperature	T_{ant}	290	K	Earth Noise Temperature
RX System Noise Temperature	T_{sys}	359.463	K	
Receive Bandwidth	B	20,000	Hz	
Receive Noise Power	P_n	-160.03	dBW	
Carrier to Noise Ratio	**CNR**	**25.01**	**dB**	
Link Margin	$\mathbf{M_{link}}$	**0.81**	**dB**	

6.3.2 FSK Link Budget

The results of the link budget analysis are presented in Table 6.4 below. As shown in the final line, the link margin is +0.09 dB. This means that the link does close for the target

BER of 10^{-5}, just barely. To re-iterate, this analysis is only valid for the moment in time when the satellite is at an elevation of 10 degrees.

Table 6.4: FSK Link Budget Analysis.

Parameter	**Variable**	**Value**	**Unit**	**Note**
Link Frequency	f_0	436.500	MHz	Center of sub-band
Link Wavelength	λ_0	0.686810	m	used for path loss calculation
Link Modulation				G3RUH_FSK
Required CNR	CNR_{req}	18.0	dB	for 10^{-5} BER, see Fig. 3.2
Receiver Implementation Loss	L_{impl}	1.0	dB	
Required Effective CNR	CNR_{req_eff}	19.0	dB	
Spacecraft Transmit Power	P_{tx}	0.0	dBW	1 Watt, assumption
Spacecraft Antenna Gain	G_{tx_ant}	2.0	dBi	Simple Dipole Pattern
Spacecraft EIRP	$EIRP_{tx}$	2.0	dBW	
Polarization Losses	L_{pol}	3.0	dB	Linear to Circular Pol Loss
Atmospheric Losses (2m)	L_{atmo}	0.35	dB	approximation
Ionospheric Losses (2m)	L_{iono}	0.40	dB	approximation
Slant Range	d	2198.339	km	Refer to text
Path Loss	L_{fs}	152.09	dB	
Receive Antenna Gain	G_{rx_ant}	13.18	dBi	from Section 5.2.3
Receive Signal Power	P_r	-140.66	dBW	
RX Component Noise Temp (2m)	T_{comp}	94.020	K	from Section 5.1.4
RX Antenna Temperature	T_{ant}	290	K	Earth Noise Temperature
RX System Noise Temperature	T_{sys}	384.020	K	
Receive Bandwidth	B	20,000	Hz	
Receive Noise Power	P_n	-159.75	dBW	
Carrier to Noise Ratio	**CNR**	**19.09**	**dB**	
Link Margin	$\mathbf{M_{link}}$	**0.09**	**dB**	

6.4 Removing The Elevation Constraint

In Chapter 4 a 10 degree minimum elevation requirement was imposed on the simulation when calculating network access time. In Section 6.3, link budget analyses for AFSK_FM and G3RUH_FSK are presented, both of which still have the 10 degree elevation constraint. However, with the use of digital modulation schemes and forward error correction, this constraint no longer makes sense. The link budget simulations presented below remove the 10 degree minimum elevation requirement and show why the constraint *is valid* for AFSK and G3RUH FSK modulations but is *no longer required* for BPSK with Rate $\frac{1}{2}$ Convolutional Coding FEC.

The simulations conducted in this section make use of the Azimuth, Elevation, and Range (AER) data exported from STK during the one month network simulation with all 19 nodes of the ground station network. This data is produced for each node in the network, for every pass of the simulation period. When a pass occurs over a ground station node (again, with no elevation constraint), data is generated for the entire pass at 60 second intervals. The combination of this data, along with the link budget equations above, allows for simulations to be generated that ultimately allow Link Margin to be compared against the elevation of the spacecraft. Again, when the link margin is positive, the link is closed, and when the link margin is negative the link is no longer closed.

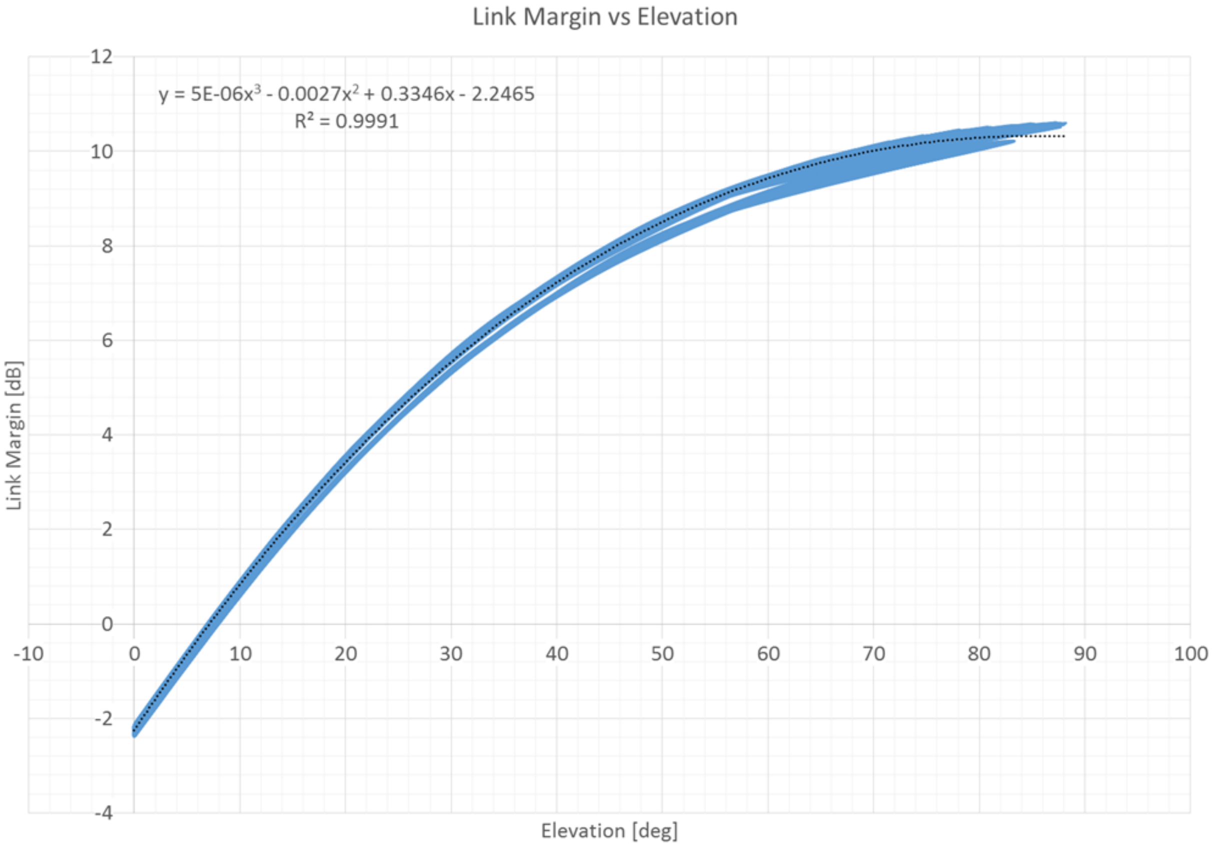


Figure 6.1: AFSK, 2m, Yagi Link Margin vs Elevation.

The results of the AFSK_FM simulation are presented in Figure 6.1. This plot shows the link margin as a function of elevation. As can be seen in the figure, the link margin becomes positive at just under 10 degrees elevation at around 7 degrees. This validates the 10 degree constraint from the original network simulations.

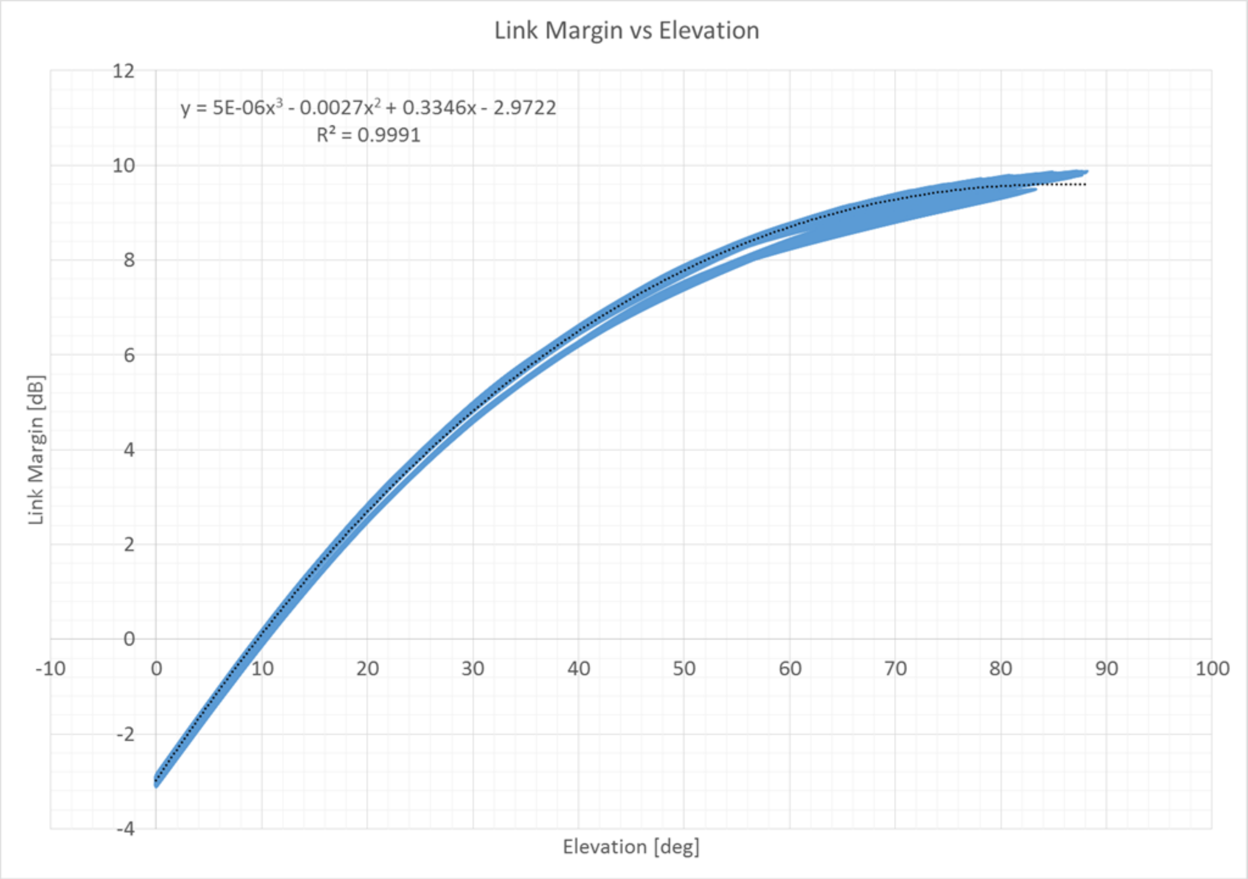


Figure 6.2: FSK, 70cm, Yagi Link Margin vs Elevation.

The results of the G3RUH_FSK simulation are presented in Figure 6.2. This plot shows the link margin as a function of elevation. As can be seen in the figure, the link margin becomes positive just at 10 degrees elevation. This also validates the 10 degree constraint from the original network simulations.

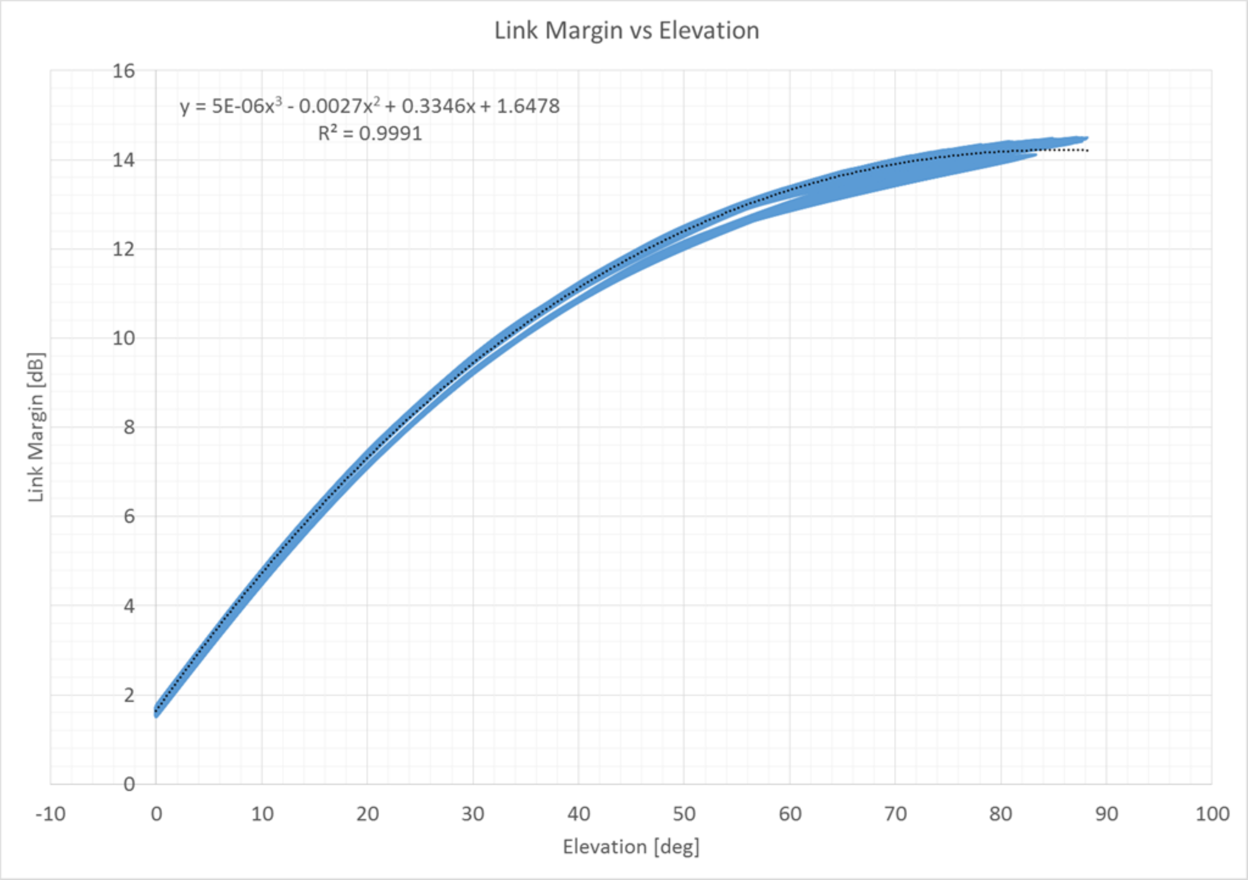


Figure 6.3: BPSK, Conv Code, R=$\frac{1}{2}$, K=7, 70cm, Hemispherical Link Margin vs Elevation.

For the following simulation, the modulation scheme used is BPSK with R=$\frac{1}{2}$, K=7 convolutional coding forward error correction. For this simulation scenario, the frequency of operation is in the center of the 70cm Amateur Satellite Service sub-band. A significant difference from the previous two simulation scenarios is the use of an omni directional antenna pattern. It was shown in Section 3.2 that switching to digital modulation schemes with forward error correction offers significant gains over AFSK and FSK modulation schemes in the link budget. Therefore, the receive antenna system used for this simulation is approximated with an antenna pattern that is hemispherical, circularly polarized, and has a gain of approximately 4.0 dBi. This pattern can be produced by such antennas as the Lindenblad or the Crossed Loop antenna (a.k.a. the Eggbeater Antenna).

Figure 6.3 shows the *significant* impact of switching to digital modulation schemes. The link margin is positive throughout the entire pass. Not only is the link closed as soon as the satellite rises over the horizon, the need for tracking is completely removed. Simultaneously, access time is increased, the robustness of the ground station node is improved without the need for mechanical steering (which would eventually fail and require servicing), and ultimately data volumes are increased.

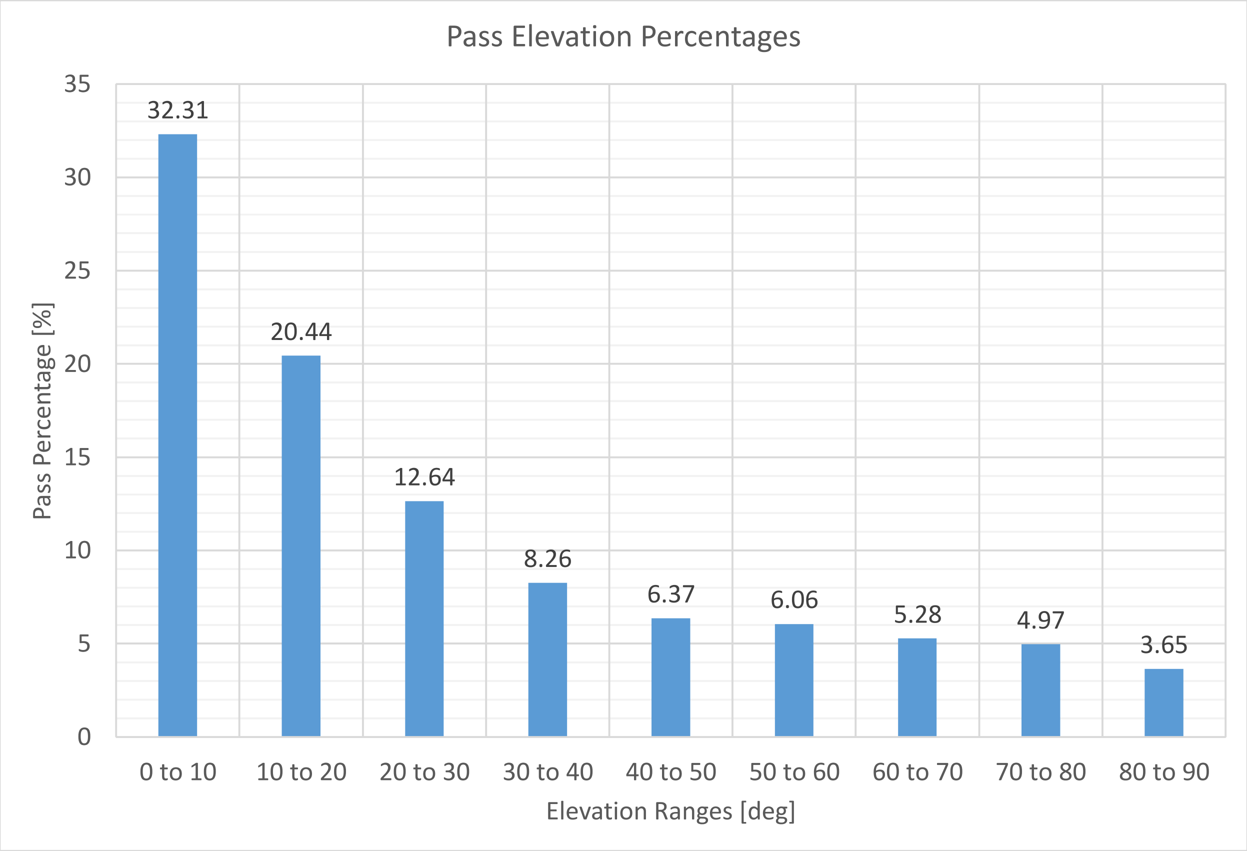


Figure 6.4: Pass Maximum Elevation Percentages.

Figure 6.4 above shows a breakdown of how many passes occur at low elevation angles. The counts are based on the maximum elevation angle achieved per pass. Just over 32% of the passes occur at elevation angles that achieve a maximum of 10 degrees elevation or less. This is a significant amount of access time that is not part of the original network simulation in terms of network access time calculation.

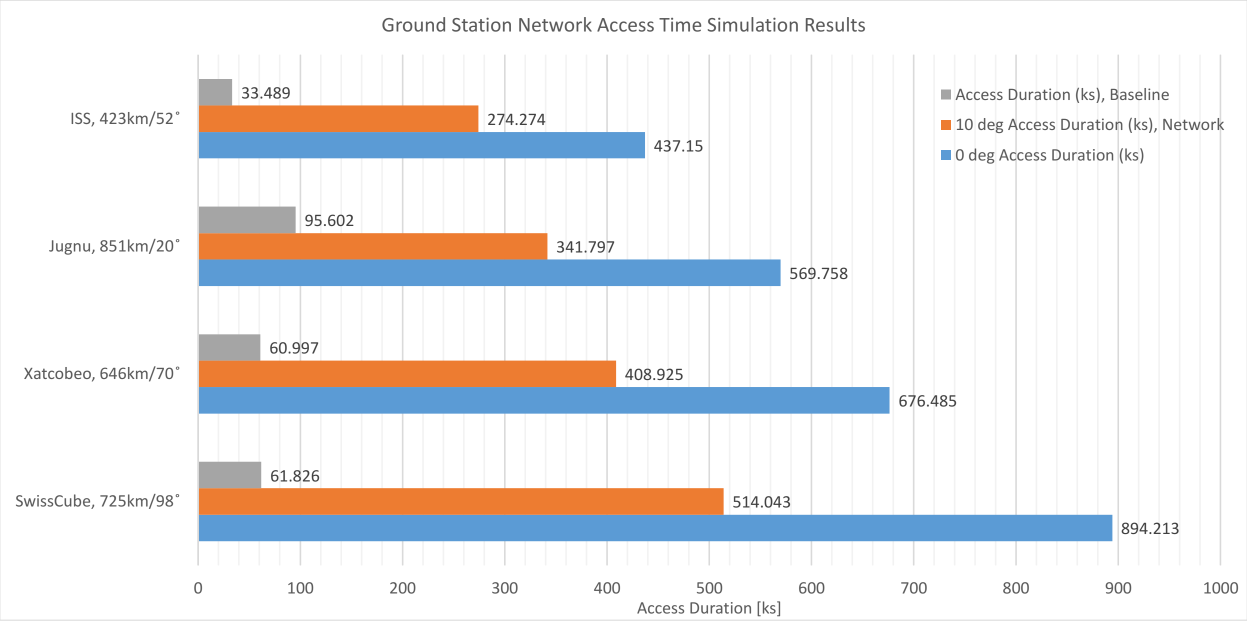


Figure 6.5: Network Access Time Simulation Results, No Elevation Constraint.

Figure 6.5 shows network access times for three scenarios. In gray, the access duration for the baseline control facility for the four simulated spacecraft are shown. Shown in orange, is the network access time with the ten degree elevation constraint. Finally, shown in blue is the access time with the 10 degree elevation constraint removed. As expected total access time, is significantly increased. For the ISS type orbit, network access times are increased by a factor of 13.05 over the control facility baseline. For the Jugnu type orbit network access times are increased by a factor of 5.96 over the baseline. For the Xatcobeo type orbit, network access times are increased by a factor of 11.09 over the baseline. For the SwissCube type orbit, network access times are increased by a factor of 14.5.

6.5 Global Distributed Ground Station Network

It has been shown that switching from the current analog modulation schemes to digital modulation schemes with the implementation of forward error correction has significant benefits for cubesat communications. Specifically, with the decreased CNR requirements, the need for tracking antenna systems is eliminated, instead allowing for antenna systems with hemispherical antenna patterns to be utilized. Similarly, the need for a 10 degree elevation limit no longer makes sense, and since a significant percentage of LEO passes occur below 10 degrees, access times are further increased.

A simulation is presented in Chapter 4 that shows the benefits of a distributed ground station network which makes use of 19 ground station nodes, mostly located in North America, with a few other nodes positioned in strategic locations (such as McMurdo Station, Antarctica). The next step is to determine the benefits of a *global* distributed ground station network. The following simulation places nodes in fifty locations around the world. The original 19 nodes are included, with an additional 31 nodes placed around the world. Capital cities of various countries are used as a guide for placing the nodes, however some are strategically positioned to fill gaps in the network coverage.

This simulation assumes that BPSK with Convolutional Coding with $R=\frac{1}{2}$, K=7 is utilized. Additionally, it also assumes that a Lindenblad antenna, with a hemispherical pattern, is also utilized. Figure 6.6 below shows the results of this simulation. As in the previous simulations, where overlap has occurred, it is removed from the calculation in order to present Network Access Time. The legend for Figure 6.6 requires some explanation. The yellow colored "Baseline, 10 deg" entries are the results of the single cubesat to its control facility simulation, for one month, with a ten degree elevation constraint in place. The gray colored "Network V1, 10 deg" entries are the results of the original 19 node network simulation, for one month, with the ten degree elevation constraint in place. The orange colored "Network V1, 0 deg" entries are the results of the original 19 node network simulation, for one month, with no elevation constraint in place. Finally, the blue colored "Network V2, 0 deg" entries are the results of the 50 node globally distributed ground station network, for one month, with no elevation constraint in place.

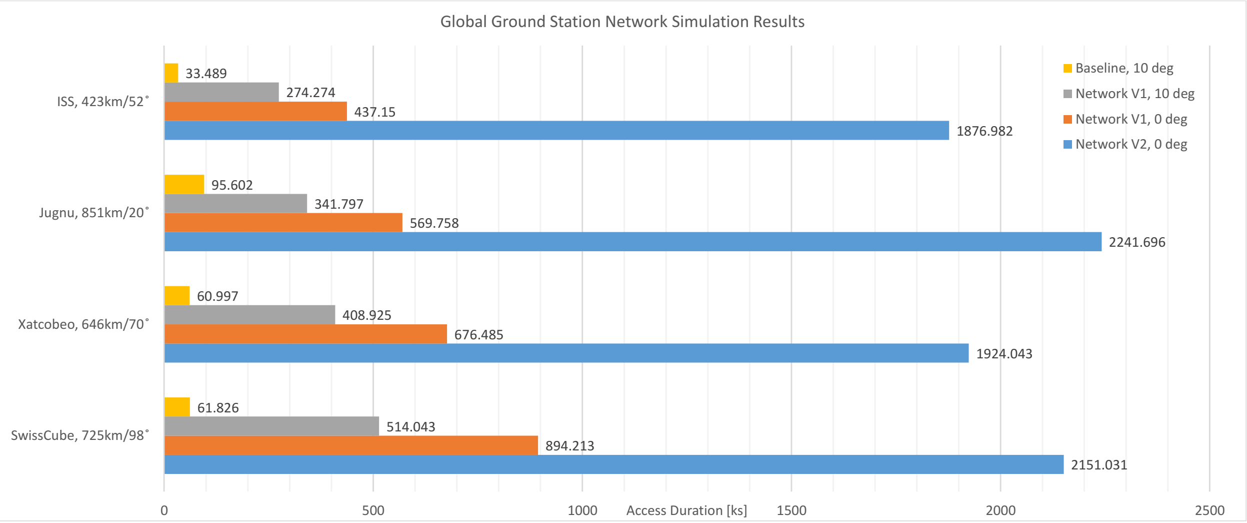


Figure 6.6: Global Network Access Time Simulation Results.

By increasing the number of ground station nodes from 19 to 50, it is shown that *significant* access time gains can be achieved. The ideal scenario would allow for 24 hours per day network access to cubesats in Low Earth Orbit. For an ISS type orbit, 16.82 hours per day can be achieved. For a Jugnu type orbit 20.09 hours per day can be achieved. For an Xatcobeo type orbit, 17.24 hours per day can be achieved. For a SwissCube type orbit, 19.27 hours per day can be achieved. This significant increase in access time clearly shows the overall benefit of a globally distributed network of ground station nodes. As a final additional note, the estimated cost of a single ground station node prototype is approximately \$2,000.00. For 50 nodes this equates to approximately \$100,000.00. This estimated cost does not account for infrastructure costs such as the central processing server and the costs for network connectivity of the nodes. Even in light of this, this cost is trivial compared to most professional grade single ground station installations.

Chapter 7

Future Work

7.1 Ground Station Node Design

7.1.1 Physical Design Improvements

A number of improvements can be made to the existing ground station design. Two notable physical problems have been observed with the current design. The first concerns weatherproofing. The prototype ground station node that has been constructed with an eye towards keeping costs low as this was an unfunded project. Therefore the enclosures for the subsystems are not weatherproof. To counteract this, the node is currently being operated underneath a 9 foot by 9 foot pop up tent, acting as a sort of poor man's radome. This is not acceptable for an actual deployment. Steps need to be taken in order to properly weatherproof the system, including weatherproof enclosures, UV resistant cabling, and wrapping of all external connections with weatherproof tape.

Additionally, an interesting, though problematic, phenomenon has been observed after sunset. The current prototype node is located near the town of Blacksburg in the mountains of south west Virginia. The node was first placed into operation in the Fall Season. Operation of the node will continue into the Winter months. Temperatures at night drop to near freezing (sometimes below) during the Fall and well below freezing during the Winter. As the temperature drops, remote access to the node ceases. This phenomena is not immediate, but a slow process that begins shortly after sunset. First, while access to the node via ssh is still possible, the FunCube Dongle Pro Plus stops functioning, and effectively does not turn on. As temperatures continue to drop, eventually a point is reached where even the Beagle Bone Black no longer operates, and remote operation of the node ceases. This is unacceptable for a real deployment of the system. This problem has not been thoroughly investigated to date, and the cause of these issues is not completely understood. The exact cause of this is not fully understood yet, i.e. is it the FCDPP and BBB themselves that are ceasing to function properly, are the voltages on the power supplies dropping with temperature, or is something else causing the issue? All that is known for sure is that when the temperature drops to near freezing and below, function of the node ceases. Temperature controlled enclosures are available to house the subsystems and will likely need to be used for a real deployment, especially if McMurdo Station, Antarctica and Anchorage, Alaska are optional node locations.

Finally, transmit capabilities would be an interesting feature to add to the design. From a technical standpoint, the addition of transmit capabilities would require a significant redesign of the system. First, a new Software Defined Radio that can properly operate with the Embedded Computer would need to be selected, since the FCDPP is a receive only device. Power amplifiers would need to be included as well as transmit/receive selection relays. The selection relays would be required in order to ensure proper isolation between the transmit and receive chains in the system in order protect the receive chain from overloads if signals are transmitted. The addition of transmit capabilities would also require a more coordinated effort with the central server. No longer would the nodes be able to operate semi-autonomously to collect data. If signals are actually transmitted back to the spacecraft coordination with a central server would be required to ensure duplicate transmissions are not made and that interference does not occur.

The addition of transmit capabilities also introduces potential legal issues. A license from the FCC or NTIA would be required in order to actually transmit. If the nodes were utilized for both cubesat operations as well as operation with Amateur Radio Satellites this introduces a significant problem. Cross service communication is not permitted by the FCC. If the Cubesat is licensed under part 5, or Experimental Radio Service, rules and the ground station nodes are licensed under Part 97, or Amateur Radio Service, rules then a conflict would exist. Further research into the proper legal licensing of the ground station nodes would be required before transmit capabilities are added to the system.

7.1.2 Automation

The current prototype node is not fully automated. Remote operation of the node is required in order to track spacecraft and receive data. For an actual deployment of the system, continued software development would be required in multiple areas. First, automatic transfer of recorded data files would be required. This is relatively simple to accomplish through automated scripting. Automatic tracking is possible with the node, but again requires remote access. Currently, an application is run on the remote computer that accesses `predict` for azimuth and elevation information and then passes this information to the tracking subsystem. This should also be relatively easy to accomplish with a python program running on the BBB.

Additionally, the scheduler needs to be further developed. The scheduler would be responsible for prioritizing the list of spacecraft that would be tracked by a given node. This scheduler would have to resolve any conflicts between concurrent passes of spacecraft over a single node, based on the priority assigned to each spacecraft in the target list. Again, this is done “manually” in the current prototype by the operator. Automation of this would be relatively complex.

Better development of the central server would also be required for a future deployment. Only a single ground station node has been developed, and it is not fully autonomous, requiring a human operator to remotely access the system to track spacecraft, execute waveforms, etc.. Similarly, the central server processing and data warehousing functionality is not fully developed. Removal of the human in the loop would require further research to fully develop the required software for the central server.

7.2 Network Capabilities

7.2.1 Orbit Determination

Since the planned network has large spatial diversity, there exists the possibility of using the network to perform orbit determination. Consider the scenario where multiple ground station receiver nodes are in view of the spacecraft. When transmission occurs from the spacecraft there will be different Doppler frequency shifts at each node. Additionally, the time of arrival of the signal will be different for each node, since the slant range between the spacecraft and each node in view will be different. This information can be used in conjunction with frequency difference of arrival (FDOA) and time difference of arrival (TDOA) algorithms to determine the position of the spacecraft. After a series of spacecraft positions is determined in time, the orbit of the spacecraft can be calculated. In order to achieve this, each node of the network would have to have accurate frequency synchronization as well as timing synchronization. This could be accomplished with for example a GPS Disciplined Oscillator to provide both time and frequency synchronization between the nodes. Further research is required in order to determine the exact requirements of the nodes from a design standpoint. Further research is also required in order to determine the best position location algorithm to use (TDOA, FDOA, or a combination of both).

The current receiver node design incorporates the use of a tracking pedestal in order to steer the antennas to properly point them at the spacecraft. This relies on knowledge of the current orbit of the spacecraft through the use of the Two Line Element set for a spacecraft combined with the SGP4 algorithm. This inherently precludes orbit determination, because knowledge of the orbit is required in order to steer the directional antennas properly. If, however, the transition to digital modulation schemes as described in Chapter 3 were to happen, then the need for directional antennas would be removed. Prior knowledge of the orbit would no longer be required from a link budget perspective. Thus, in order for orbit determination to be a viable feature of the distributed ground station network, the utilization of digital modulation schemes by the spacecraft would also be required.

7.2.2 Signal Combining

Another possibility exists due to the large spatial diversity of the node locations. Consider the scenario of a packet that is transmitted by the spacecraft and received by multiple nodes, but *fails* to be properly decoded by all of the receiving nodes. The recorded samples could be transferred to the central processor and signal combining algorithms can be used to take the information from the multiple nodes and attempt to combine them in such a way as to be able to properly decode the packet. Signal Combining techniques already exist, but further research would be required to determine the best algorithm to use for this network.

Chapter 8

Conclusions

This works has discussed three key areas related to cubesat communications in an effort to design a distributed network of ground stations in order to maximize the amount of data that can be downloaded from orbit. First, the current state of cubesat communications is examined in Chapter 2. Trends in existing cubesat communications systems are introduced to include frequency utilization, modulation schemes, and legal licensing requirements. Chapter 3 analyzes the current cubesat communications protocols and then calls for alternative modulation schemes and the use of forward error correction in order to create more efficient communication methods. A notional design of the ground station network is then presented in Chapter 4. This network consists of 19 ground station locations distributed throughout North America and a few other locations. Results are presented that show the drastic increase in per day data volumes that can be generated by spacecraft should such a network be implemented. A candidate design for the ground station receiver node is presented that would enable the deployment of such a network is presented in Chapter 5. This candidate design has been realized in an actual prototype and has successfully tracked multiple cubesats that are currently active, thus proving the feasibility of the design. Finally, Chapter 6 presents Link Budget Analyses and Extended Network simulations that further prove the utility of switching to digital modulation schemes and deploying a globally distributed network of 50 ground station nodes. The combination of the transition to digital modulation schemes that employ Forward Error Correction, as well as the deployment of a distributed network of ground station nodes has the potential to transform modern cubesat communications.

Bibliography

[1] S. Ford. "A Brief History of Amateur Radio Satellites." *The ARRL Satellite Handbook*, CT: American Radio Relay League, 2008, 1-1.

[2] B. Klofas, "The Future of CubeSat Data Communications", in *Proceedings of AMSAT-NA Space Symposium*, Oct. 2012.

[3] M. Wakita. (2013, November 5). *All Satellites Frequency List Update.* [Online]. Available:`http://www.ne.jp/asahi/hamradio/je9pel/satslist.htm`

[4] U.S. National Archives and Records Administration. *Code of Federal Regulations.* Title 47. Telecommunications. Chapter I. Federal Communications Commission. Part 97. Amateur Radio Service.

[5] *Guidance on Obtaining Licenses for Small Satellites,* FCC DA: 13-445, March 15, 2013.

[6] AX.25 Link Access Protocol for Amateur Packet Radio. *Tucson Amateur Packet Radio.*1998. Retrieved 2013-11-05.

[7] G. Florian. (2007, March 19). *Telemetry and Telecommand Transfer Frames Format.* [Online]. Available: `http://swisscube-live.ch/Publish/S3-BC-SE-1-1b-AX.25%20Transfer%20Frames%20Format.pdf`

[8] J.A. King. (2008, March 6). *AMSAT / IARU Annotated Link Model System Version: 2.5.2.* [Online]. Available: `http://wiki.oz9aec.net/index.php/File:AMSAT-IARU_Link_Model_Rev2.5.2.ods`

[9] J. Miller, "9600 Baud Packet Radio Modem Design," in *ARRL 7th Computer Networking Conference*, US, 1988, pp. 135-140.

[10] J.A. Magliacane. (1998, March). *The KD2BD 9600 Baud Modem.* [Online]. Available: `http://www.amsat.org/amsat/articles/kd2bd/9k6modem/9k6modem.html`

[11] M. Schmidt.*Ground Station Networks for Efficient Operation of Distributed Small Satellite Systems.* Diss. Julius Maximilian University of Würzburg, 2011.

[12] J. Cutler, P. Linder, and A. Fox. "A Federated Ground Station Network," in *SpaceOps Conference Proceedings*, Oct. 2002.

[13] "GENSO". *www.genso.org*. Web. Oct. 2013.
<`http://www.genso.org/`>

[14] T. Pratt *et al.*. *Satellite Communications*. 2nd Ed. New York, NY: Wiley, 2003.

[15] H. Long. "Specifications." *FunCube Dongle Pro Plus*. Oct. 2012. Web. Oct. 2013.
<`http://www.funcubedongle.com/?page_id=1201`>

[16] "RaspberryPi" *www.raspberrypi.org*. Oct. 2012. Web. Oct. 2013.
<`www.raspberrypi.org`>

[17] "BeagleBone Black." *beagleboard.org*. Web. Jun. 2013.
<`http://beagleboard.org/Products/BeagleBoneBlack`>

[18] "GNU Radio Homepage." *GNU Radio*. Web. Oct. 2013.
<`http://gnuradio.org/redmine/projects/gnuradio/wiki`>

[19] A. Csete. "Funcube Dongle Pro and Pro+ on the Raspberry Pi." *OZ9AEC*. Web. Jun. 2013.
<`http://www.oz9aec.net/index.php/funcube-dongle`>

[20] J. Seybold, *Introduction to RF Propagation*, NJ: Wiley & Sons, 2005, Section 4.3.

[21] L. Devlin & C. Duey. "KI0AG Micro-Duplexer." *k0lee.com*. Oct. 2012. Web. Oct. 2013.
<`www.k0lee.com/duplexer.htm`>

[22] R. Straw. "Antenna Systems for Space Communications." *The ARRL Antenna Handbook*, CT: American Radio Relay League, 2000, 19-9 to 19-13.

[23] A. Csete. "GNU Radio Projects." *OZ9AEC*. Web. Jun. 2013.
<`www.oz9aec.net/index.php/gnu-radio`>

Appendix A

CubeSat Data

This section provides a list of currently active cubesats. This list was generated by a Japanese Amateur Radio Operator named Mineo Wakita, JE9PEL [3]. The list provided below is current as of November 5th, 2013.

Table A.1: Active Satellites Occupying Amateur Radio Spectrum

Common	**NORAD**	**Uplink**	**Downlink**	**Beacon**		
Name	**ID**	**[MHz]**	**[MHz]**	**[MHz]**	**Mode**	**Callsign**
AO-7 (Phase-2B)	07530	145.850-950	29.400-500	29.502	A	
AO-7 (Phase-2B)	07530	432.125-175	145.975-925	145.97	B/C	
UO-11 (UoSAT-2)	14781	-	145.826/435.025	2401.5	(V)FM,(S)PSK	UOSAT-2
RS-15 (Sputnik)	23439	145.858-898	29.354-394	29.352	SSB/CW	
LO-19 (LUSAT)	20442	145.840-900	437.125/150	437.125	CW Carrier	LUSAT-11, -12
RS-22 (Mozhayets)	27939	-	-	435.352		
IO-26 (ITAMSAT)	22826	145.875-950	435.822/867	435.791	1200bps JAS	ITMSAT-11,-12
AO-27 (EYESAT-A)	22825	145.85	436.795	436.795	1200bps AFSK,FM	
FO-29 (JAS-2)	24278	145.900-999	435.900-800	435.7964	SSB,CW	8J1JCS
RS-30 (Yubileiniy-1)	32953	-	-	435.215/3	15 CW	
RS-30 (Yubileiniy)	32953	-	435.315/215	435.315	CW	RS30
GO-32 (TechSat)	25397	145.850/930	435.225	435.225/325	9600bps FSK	4XTECH-11,-12
ISS (ZARYA)	25544	145.2	145.8	-	Voice (Reg 1)	NA1SS
ISS (ZARYA)	25544	144.49	145.8	-	Voice (Reg 2,3)	NA1SS
ISS (ZARYA)	25544	145.99	145.8	145.8	Packet	RS0ISS
ISS (ZARYA)	25544	437.55	437.55	437.55	APRS	RS0ISS
ISS (ZARYA)	25544	145.825	145.825	145.825	APRS	RS0ISS-4, -11
RS-39 (Chibis-M)	38051	-	-	435.215/315	CW	
NO-44 (PCsat1)	26931	145.827	145.827	145.827	1200bps AFSK	(A) PCSAT-1
NO-44 (PCsat1)	26931	435.25	145.827	145.827	9600bps FSK	(A) PCSAT-2
NO-44 (PCsat1)	26931	-	144.39	144.39	1200bps AFSK	(B) PCSAT-11
NO-44 (PCsat1)	26931	-	144.39	144.39	9600bps FSK	(B) PCSAT-12
SO-50 (SaudiSat-1c)	27607	145.85	436.795	-	FM_tone 67.0Hz	
VO-52 (Hamsat)	28650	435.220-280	145.930-870	145.936	SSB,Carrier	Indian
VO-52 (Hamsat)	28650	435.225-275	145.925-875	145.86	SSB,CW	Dutch
CO-55 (CUTE-I)	27844	-	437.47	436.8375	1200bps AFSK	JQ1YCY
CO-57 (XI-IV)	27848	-	437.49	436.8475	1200bps AFSK,CW	JQ1YCW
CO-58 (XI-V)	28895	-	437.345	437.465	1200bps AFSK,CW	JQ1YGW
CO-65(CUTE1.7+APDII)	32785	1267.6	437.475	-	9600bps GMSK	JQ1YTC
CO-65(CUTE1.7+APDII)	32785	-	437.475	437.275	1200bps AFSK,CW	JQ1YTC
CO-66 (SEEDS-II)	32791	-	437.485	437.485	1200bps FM,CW,Talk	JQ1YGU
COMPASS-1	32787	-	437.405	437.275	1200bps AFSK,CW	DP0COM
AAUSAT-II	32788	-	437.432	437.432	1200bps FFSK/MSK	
AAUSAT-II	32788	-	437.432	437.432	9600bps FSK	OZ2CUB
DO-64 (DELFI-C3)	32789	-	145.87	145.867	1200bps BPSK	DLFIC3
PRISM (HITOMI)	33493	-	437.425	437.25	AFSK,GMSK,CW	JQ1YCX
STARS(KUKAI mother)	33498	-	437.485	437.305	1200bps FM,CW	JR5YBN
STARS(KUKAI daughter)	33498	-	437.465	437.275	1200bps FM,CW	JR5YBO
KKS-1 (KISEKI)	33499	-	437.445	437.385	1200bps AFSK,CW	JQ1YYY
SwissCube-1	35932	-	437.505	437.505	1200bps BFSK,CW	HB9EG1
BeeSat	35933	-	436.000	436.000	9600/4800bps GMSK	DP0BEE
ITU-pSat1	35935	-	437.325	437.325	19200bps GFSK,CW	
HO-68 (XW-1)	36122	145.925-975	435.765-715	435.79	SSB inverting,CW	BJ1SA-11,-12
HO-68 (XW-1)	36122	145.825	435.675	435.79	FMtone67Hz,CWonly	BJ1SA-11,-12
TIsat-1	36799	145.98	437.305	145.98	FM,AFSK,PSK,CW	HB9DE
JUGNU	37839	-	437.505	437.2759	CW	
SRMSAT	37841	145.9	437.500	437.425	CW	
AO-71 (AubieSat-1)	37854	-	437.475	437.473	1200bps AFSK,CW	
MCubed & EXP-1 Prime	37855	-	437.505	437.502	1200bps AFSK,LSB	
MCubed & EXP-1 Prime	37855	-	437.485	437.485	9600bps GMSK,KISS	
MO-72 (MaSat-1)	38081	-	437.345	437.345	625/1250bps GFSK,CW	HA5MASAT
XATCOBEO	38082	-	437.365/145.940	437.365	1200bps FFSK,SSR,CW	
AAUSAT-3	39087	-	437.425	437.425	4800bps FSK,CW	OZ3CUB
STRAND-1	39090	-	437.568	437.568	9600bps GMSK	
AIST-2	39133	-	435.215	435.265	FM,CW	
SOMP	39134	-	437.485	437.503	1200bps AFSK,CW	DP0TUD
BeeSat-2	39136	-	435.95	435.95	4800bps GMSK,CW	DP0BEF
BeeSat-3	39135	-	435.95	435.95	4800bps GMSK,CW	DP0BEF
CUBEBUG-1	39153	-	437.445	437.4383	1200bps AFSK	CUBEB1
ESTCUBE-1	39161	-	437.505/2401.250	437.254	9600bps GMSK,CW	ES5E-11,ES5E/S

Table A.2: Upcoming Cubesats (Manifest for launch in November 2013)

Common Name	**Uplink [MHz]**	**Downlink [MHz]**	**Beacon [MHz]**	**Mode**
Black Knight-1	-	437.345	-	-
CAPE-2	-	145.825/437.325	-	-
COPPER	-	437.290	-	9600bps
DragonSat-1	-	145.870	-	
Ho’oponopono-2	-	427.220	-	9600bps FSK/GMSK
KYSat-2	-	437.405	-	
NPS-SCAT	-	437.525/2401.20	2447.600	
PhoneSat-v2	-	437.425/2401.20	2431.200	
SwampSat	-	437.385	-	
TetherSat	-	437.100/305	-	9600bps GFSK
TJ3Sat	-	437.320	-	
Trailblazer-1	-	437.425	-	
BRITE-PL1	-	437.xxx, 2.2GHz	-	
CubeBug-2	-	437.445	-	1200bps AFSK, 9600bps FSK
Delfi-n3Xt	435.530-570	145.880-920	-	Transponder(U/V)
Delfi-n3Xt	-	145.870/930	-	1200bps AFSK
Eagle-1	-	437.465	-	9600bps GFSK
Eagle-2	-	437.505	-	9600bps GFSK
E-Star-2	-	437.485	-	1200bps AFSK
First-MOVE	435.520	145.970	-	1200bps BPSK
FUNcube-1	435.150-130	145.950-970	-	Inverting(U/V)
FUNcube-1	-	145.935	-	1200bps BPSK
GOMX-1	-	437.250	-	1k2/2k4/4k8/9k6 GMSK
HiNCube	-	437.305	-	
Humsat-D	-	437.325/437.525	-	
ICUBE-1	435.060	145.947	-	1200bps BPSK
NEE-02 Krysaor	-	910	-	
PUCP-SAT-1	145.840	145.840/437.200	-	1200bps AFSK
Pocket-PUCP	-	437.200	437.200	1200bps AFSK,CW
Qubescout-S1	-	437.525	-	9600bps GMSK
Triton-1,2	435.xxx	145.815/860	2408	9600bps / RC-BPSK
UniSat-5	-	437.175/425	-	9600bps GMSK
UWE-3	-	437.385	437.385	1200bps FSK,CW
VELOX-P2	437.305	145.980	-	1200bps BPSK
Wren	-	437.405	437.405	1200bps FSK,CW
ZACube-1	145.860	437.345	14.099	
PicoDragon	-	437.365	437.250	1200bps AFSK,CW
ArduSat-1	-	437.325	-	9600bps MSK
ArduSat-X	-	437.345	-	9600bps MSK
TechEdSat-3	-	437.465	437.465	1200bps AFSK,CW