The Lightspeed Rangefinder

By

Jamie Smith

Researched At

USC Rocket Propulsion Lab

Submitted to the University of Southern California in partial fulfillment

of graduation requirements for honors degree

Viterbi School of Engineering

May 2021

Research Advisor: Dr. Dan Erwin

Signature:______________________

Honors Advisor: Prof. Andrea Armani

Signature:______________________

UNIVERSITY OF SOUTHERN CALIFORNIA

VITERBI SCHOOL OF ENGINEERING APPROVAL

of an honors undergraduate thesis submitted by

Jamie Smith

This thesis has been reviewed by the research advisor, engineering honors research advisor, and honors program director, and it has been found to be satisfactory.

Prof. Dan Erwin, Research Advisor

Prof. Andrea Armani, Honors Advisor

Prof. Sandeep Gupta, Program Director

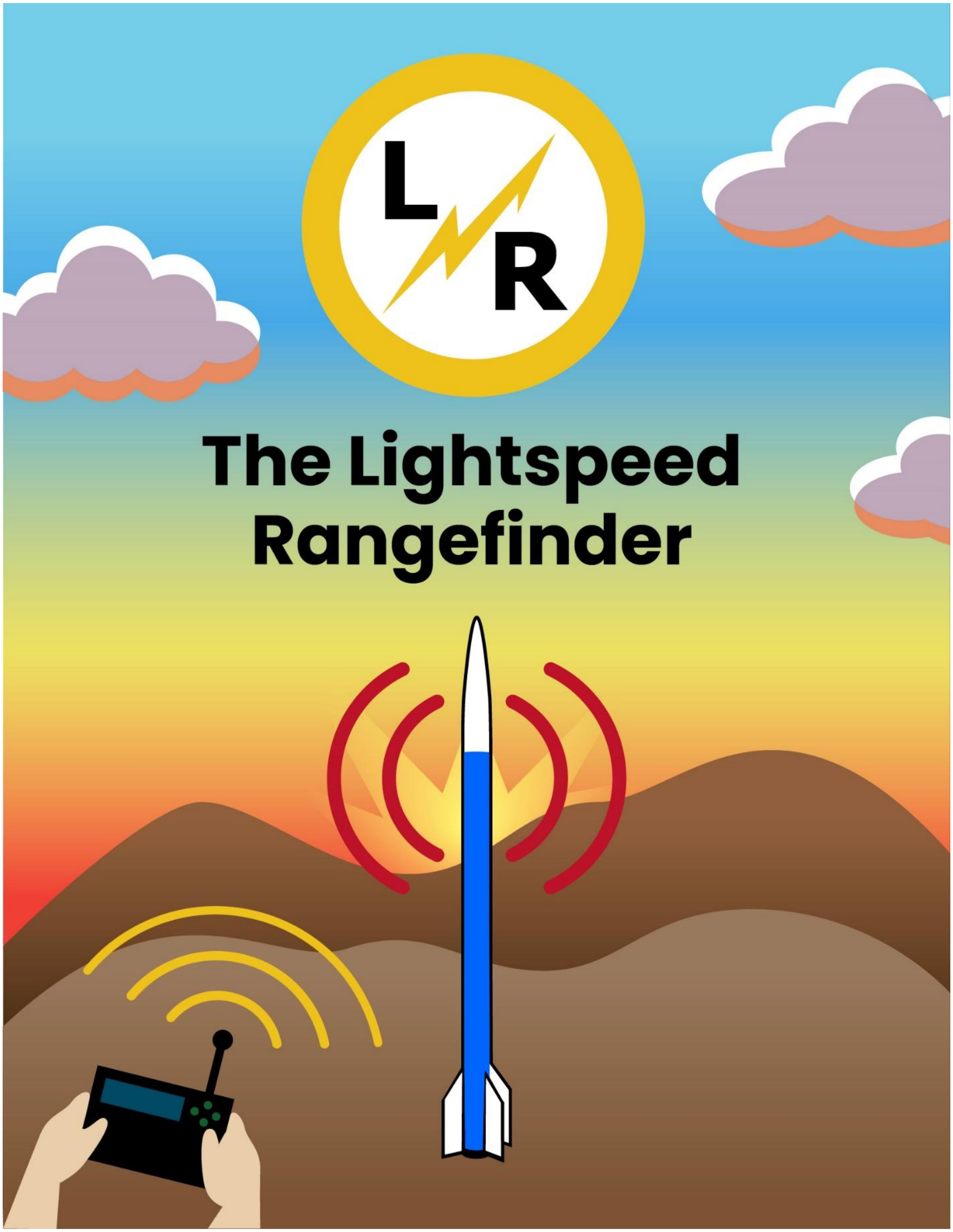

L
R
The Lightspeed
Rangefinder

Table of Contents

Abstract

The Lightspeed Rangefinder is a novel application of a radio-based distance measuring system built from consumer components to the field of amateur rocketry. Using an arrangement where multiple ground stations communicate with a transponder attached to a flight vehicle, the vehicle's altitude can be determined to within roughly 100 meters at altitudes of over 100 km. While each individual measurement is imprecise, the system runs at a high polling rate so averaging allows a more precise position to be produced from a large number of less precise points. This system has major implications for Rocket Propulsion Lab's future operations, and we believe that it will allow the flight path of its vehicles to be tracked much more accurately and reliably than was ever possible before. Over the last year and a half, a team of engineers at the USC Rocket Propulsion Lab (led by Jamie Smith) conceptualized the rangefinder system, designed and built multiple iterations of PCBs, created embedded software to run the system, ran a bevy of qualification tests, and prototyped methods of analyzing the resultant points. While the first flight test of the system was inconclusive, long-distance ground tests of the rangefinder produced very promising results. This thesis will detail the process of design, iteration, and revision that led to the rangefinder system, as well as the theory by which it operates and how the data it produces is analyzed.

Acknowledgements

The following members of Rocket Propulsion Lab brought the rangefinder from a concept to a reality!

- Jamie Smith: Project lead, designed initial PCBs, wrote & debugged CC1200 driver and ranging firmware, performed RF lab testing
- Matthew Ai: Contributed to analysis code and firmware, ran launch day ops, assisted with PCB design and routing
- Lina Al Adli: Helped write analysis code and firmware, assisted with PCB design and routing
- Anna Shaposhnik, Kevin Moran: Designed housing for ground stations
- Luke St. Regis: Suggested initial implementation of analysis code
- Lauren Potterat: Coordinated launch activities with bps.space
- Raphael Navarro: Designed power supply schematics
- Arpad Kovesdy: Assisted in initial concept development
- Conor Hayes: Commissioned initial research project
- Ground Station Operators:
  - Jackie Hall (x2)
  - Jay Sridharan
  - Mark Stephenson
  - Aaron Harris
  - Raphael Navarro
  - Ava Badii
  - Faheem Chunara
  - Kate McQuarrie
  - Jess Ridgeway
  - Audrey Roberts

We would also like to thank:

- Joe Barnard and the folks at bps.space, for giving us a chance at a high-altitude flight in turbulent times

- Prof. Aluizio Prata, who gave us some great preliminary feedback on the writeup
- Randall Smith and Stacy Bermingham, for physical and emotional support in a hundred different ways

Chapter 1: Background

Tracking amateur rockets is a difficult problem. Across the wide open spaces of the United States, thousands of amateur engineers and scientists practice model rocketry as a hobby. People design and build all sorts of different designs, from miniature sugar rocket kits to high-power custom rockets that launch to tens of thousands of feet into the air. However, all of these people face a common problem: tracking how high these rockets reach and how well they perform. There are a number of existing solutions to this problem, such as GPS trackers and barometric altimeters, but each is subject to a number of limitations. Some do not produce accurate data at high speed, and almost none are very accurate at high altitudes.

Figure 1.1. Tripoli amateur rocketry event. http://uscrpl.blogspot.com/

USC Rocket Propulsion Lab (USC RPL) experienced this dilemma firsthand on our Traveler 4 launch in April 2019. This was the fifth of USC RPL's space shots, but the first one to successfully reach over 100 km and cross the Kármán line into space. However, due to the hypersonic flight, strenuous acceleration conditions, and high apogee, many of the sensors on

Traveler 4's avionics unit (including the GPS and the barometric altimeters) were rendered useless for determining apogee. In the end, we had to perform a complex analysis involving double-integrating acceleration data after applying rotations from the gyroscope and magnetometer. Thankfully this method was effective, and we determined the vehicle's apogee as 103.5km. However, integrating already-imprecise accelerometer data magnified its error even further, and we ended up with mammoth error bars of +-4km on our apogee [1]. USC RPL realized that action needed to be taken to fix this by our next launch, and settled on a multi-pronged approach involving both integrating new, more accurate sensors and researching alternate methods of measuring our apogee. The system here presented is the result of that research.

In this paper we, the students of USC Rocket Propulsion Lab propose a novel system that can track rockets without being subject to any of the limitations of other methods: the Lightspeed Rangefinder. By measuring the propagation time of radio signals, our system records the distance from several ground stations to a transponder mounted inside a rocket. Then, by processing this data, it computes the rocket's position in three-dimensional space at each point in time. This approach sidesteps the limitations of GPS and other tracking methods and allows the rangefinder to produce quality data even under adverse conditions such as high altitudes and supersonic velocities. Thanks to its small size, accuracy at distance, and versatile nature, we believe that the rangefinder will be a vital tool for RPL's future space shots as well as for the larger rocketry community.

Comparison to Other Systems

There are currently three types of rocket tracking devices commonly used by amateurs and collegiate rocketry teams: altimeters, accelerometers, and GPSs. Each of these can provide useful data, but is also affected by a number of limitations. In this section we will provide an overview of what these are, and how the rangefinder may be able to perform better.

Altimeters are the most common type of tracking device, and can be made extremely compact and simple. They measure air pressure, from which altitude can be calculated using standard atmospheric models. However, they have several limitations. First and foremost, they can only track movement in the vertical axis. This means they aren't useful for analyzing horizontal movement, or for finding a rocket after it has landed. Their accuracy is acceptable, but not perfect: one common sensor has an accuracy of +-0.15 kPa [2], translating to about +-15 meters at sea level. Weather conditions that differ from the standard atmospheric pressure model can also increase this error. Altimeters are also vulnerable to error due to aerodynamic effects during a rocket's flight, since supersonic airflow can artificially raise the air pressure inside the nose cone [1]. This error is difficult to quantify and can lead to data from a rocket's ascent being unusable. Worst of all, altimeters are limited in the altitude they can measure. At a certain point (it depends on the altimeter, but often between 50,000 and 100,000 ft), the air becomes so thin that no measurable air pressure at all can be sensed, and the rocket's altitude cannot be determined. Altimeters are usable for low-altitude flights as long as only basic data is needed and supersonic speeds are not reached, but they are not adequate at all for high-altitude and high-power flights.

Accelerometers are another common sensor for rocketry applications, and they are commonly used together with altimeters in consumer devices. Accelerometers work by sensing g-forces acting on the rocket and are useful because they can help capture small vibrations and speed variations that other sensors would miss. By themselves they can only be used to calculate one-axis height, and only during the ascent phase of the launch where the rocket is pointing straight up. However, when combined with another sensor, an inertial measurement unit (IMU), they can also be used to calculate position in three dimensions by applying information about the rocket's orientation [1]. Unfortunately, to calculate position the acceleration data must be double-integrated. Consumer-grade accelerometers are not very accurate; at best, they can reach the +-0.1g range. Furthermore, as we saw on Traveler 4, the double-integration magnifies error in the acceleration data significantly. So, while accelerometers have no altitude limit, they are inherently inaccurate and produce low-quality data.

The final type of sensor, and the one that is most common on more advanced rockets, is the Global Positioning System (GPS) unit. GPS-based trackers receive signals from satellites orbiting the Earth and, through an extremely complex processing method, lock on to these signals, measure their propagation time from the satellites, and calculate their own location in 3D space. In theory, this should provide high-quality position data, but unfortunately GPS modules are subject to some real-world limitations. To function, they must carefully monitor the satellite signals that are present and align their internal receiving logic with each satellite that is transmitting. If the signals are inconsistent or are changing too fast (as is often the case when the receiver is in an accelerating and spinning rocket) it can disrupt this process, reducing the GPS module's accuracy greatly or causing it to lose lock completely. This is difficult to model or predict in advance, meaning that GPS-based rocket tracking is always something of a roll of the dice as to whether the module can maintain lock. Even worse, the government enforces limitations known as a COCOM lockouts, where consumer GPS units must disable their output if they exceed a certain speed or velocity (the specifics depend on the GPS). This is to prevent their use in guided missiles, but unfortunately also impacts amateur rocketry. While there exist specialized aerospace and satellite GPS units that can overcome these limitations, these are restricted as if they were weapons and are not really possible for an amateur to purchase.

Many GPS trackers also feature radio telemetry, which allows a ground station at the launch site to receive the rocket's position during the launch. This is useful to many flyers because it helps them locate their rockets after they land, something which normally requires lots of searching and a bit of luck. Telemetry is also useful in the case that the rocket breaks up midair or fails to deploy its parachute, which can often lead to the destruction of its avionics unit. Analysis of the telemetry can provide insight into what made the rocket fail and where its pieces may have ended up.

The Lightspeed Rangefinder we have designed has clear advantages over these existing systems. Since it triangulates using radio alone, it is not affected by government lockouts or poor GPS signals -- it works with rockets regardless of their velocity, and it can operate at distances up to its own maximum radio range, over 250 km. While the rangefinder is not quite as accurate numerically as an altimeter or GPS, it is still sufficient for rocketry applications, and its accuracy does not decrease at all no matter how far the rocket travels. Additionally, since all of the rangefinding data is stored on the ground, it's safe from loss in a rocket crash, making the rangefinder equivalent to the very best telemetry systems available. All considered, the Lightspeed Rangefinder provides a very attractive alternative to these existing systems. Now, we will explain how it works.

Chapter 2: How the Ranging Works

Basic Process

Measuring distance using radio may sound complicated, but at its core, the process relies on basic laws of physics. The rangefinder operates through a simple ping-response process, as illustrated in Figure 2.1.

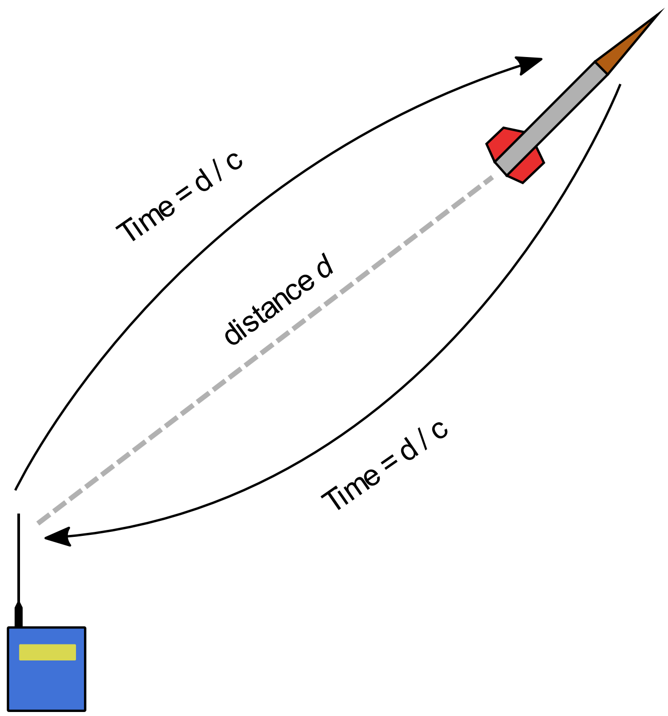


Figure 2.1. Rangefinder Operation.

First, the ground station sends a radio signal up to the transponder mounted on the rocket. The radio signal moves at the speed of light, so it takes time $\frac{d}{c}$ to reach its destination, where c is the speed of light and d is the distance from the ground station to the rocket. Then, the transponder unit inside the rocket sends a response transmission, which takes the same amount of time to travel back to the ground station. The total time taken for this process is $2\frac{d}{c} + k$, where k is a constant amount of delay created by the radio system. k can be calibrated out since it is constant, and since the speed of light is constant, the ground station can calculate the distance to the rocket by measuring the time taken for this process.

However, while this concept may be simple, the design of the actual rangefinder system is more complex, as it must address several difficult problems that emerge including accurate timing, long-distance radio communication, and the rejection of radio interference. A summary of the design we created will be given in the next section, and the details of how we arrived at this design will be explained in Chapter 3.

Solving Problems with Hardware

Radio duties on the rangefinder are handled by Texas Instruments' CC1200 RF transceiver integrated circuit. While very complex, and sometimes difficult to work with, this chip offers a huge amount of features in a very small package. Fundamentally, it operates by converting binary data sent by the CPU into shifts in the amplitude and/or frequency of a radio signal (a process called modulation). This signal is then transmitted out of its own internal amplifier. Equivalently, when in receive mode, the CC1200 listens for a radio signal to be detected matching its configured encoding. Once it finds one, and it detects a special "sync word" identifying the start of a message, it begins to decode ("demodulate") the signal into bytes. This modulation and demodulation process forms the basis of the transmissions of rangefinder system, and, under the right conditions, can transmit data over distances of several hundred kilometers.

However, to traverse these distances, analog radio signals need more power than the CC1200 alone can provide (only +14.5dBm [3]). So, additional amplifiers are needed: a Low Noise Amplifier (LNA), which amplifies received signals before they enter the CC1200, and a Power Amplifier (PA), which amplifies signals to the highest possible power before they are transmitted out of the antenna. We found several usable candidates for these amplifiers which provided hundreds of kilometers of range, and the complete details and schematics are given in Chapter 3.

With the amplifiers providing the needed radio range, the next problem becomes accurate timing. Every nanosecond of timing uncertainty translates to another 0.3 meters of distance uncertainty, so it's important to minimize error as much as possible. To record these times, we made use of the CC1200's "sync detect" output, which provides a logic high at the precise time that a radio signal's sync word arrives. This output is fed into a counter module on the rangefinder's CPU which records the precise time according to the CPU clock (80 MHz) that each pulse was received. To reduce error further, a temperature-compensated crystal oscillator (TCXO) was used to drive the CPU clock, reducing timing error to a few parts per million. Unfortunately, as explained in Test 1, we did have to accept a relatively high amount of jitter from the CC1200 itself. This limits the precision of each measurement, but by averaging these values together, a reasonably precise result can still be obtained.

Since multiple ground stations must be used at the same time, the rangefinder also faces another timing problem of a completely different kind: sharing a single radio band between multiple different ground stations. We elected to use a Time Division Multiple Access (TDMA) approach, where ground stations are only allowed to transmit pings during specific time periods. This method is easy to implement using the hardware we have, and completely removes the risk of ground stations corrupting each other's transmissions by transmitting at the same time. The only downside: ground stations must have a common, shared timebase that they can reference, which isn't easy when the ground stations in question could be miles apart. So, we elected to add a U-Blox MAX-8 GPS module to each ground station – while the goal of this system may be to avoid using GPS on the vehicle itself, there is nothing preventing its use on the stationary ground stations. This module provides a highly accurate timepulse output marking each GPS second, to which the CPU's internal clock is synchronized once a second. This provides all ground stations with a common, microsecond-accurate timebase, and also lets each ground station mark its own location so that the data can easily be processed later.

By putting these pieces of hardware together into a single unit, after fixing a lot of bugs, issues, and incompatibilities and applying an unholy amount of elbow grease, we have found that it is possible to solve the difficult problems and construct a working rangefinder.

The Measurement Process

Now that we have gone over the hardware of the system, and how it works around some of the main issues inherent in the technology, we can discuss the system at a higher level. This section will provide a low-level overview of how a ranging reading is taken using the software and digital logic on the rangefinder units. While a couple more steps have to be taken to account for the complexities of the full system, the process is still similar to the basic measurement process discussed previously.

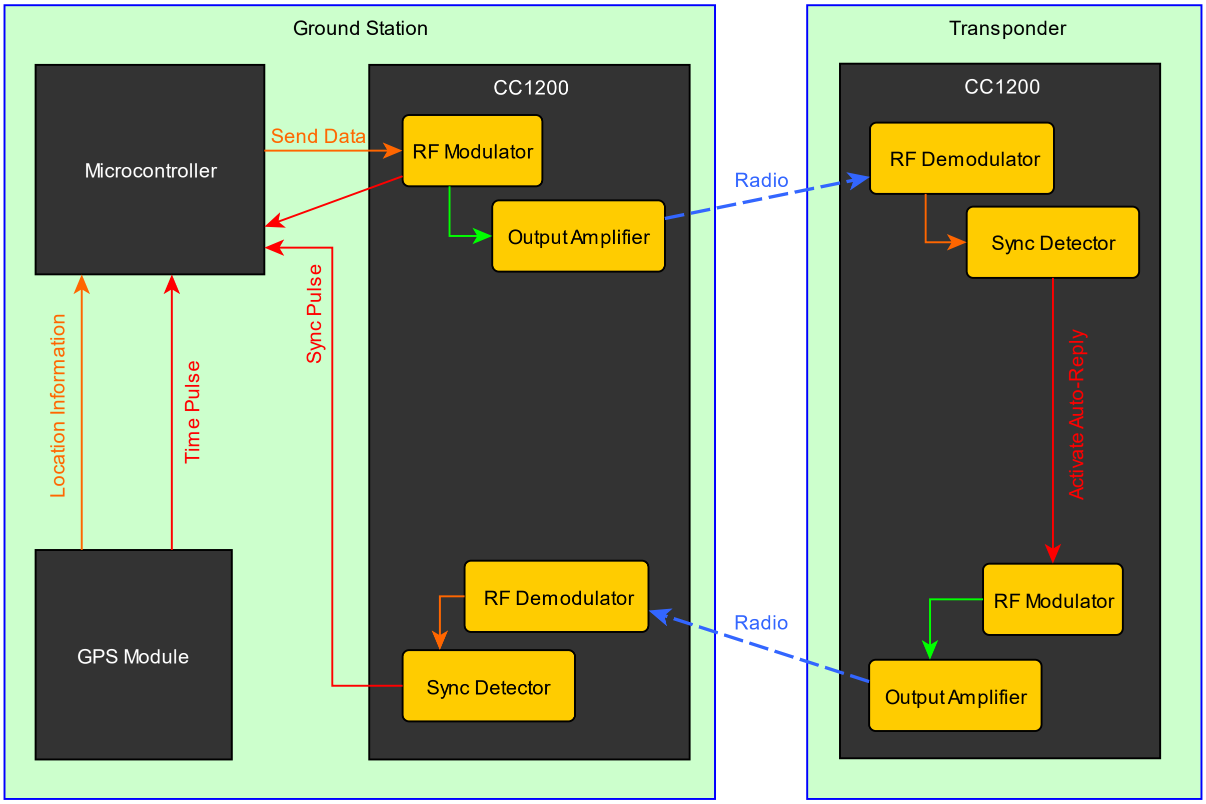


Figure 2.2. Flow of information in a single ranging measurement. RF amplifiers and other purely analog components omitted for simplicity.

1. The ground station's microcontroller decides that it is the correct time to take a measurement. Multiple ground stations coordinate their measurements according to GPS time, and the exact time and order is configurable in software on the unit.
2. The microcontroller loads some data into the CC1200 to be transmitted, then enables transmit mode.
3. The ground station microcontroller waits for a transmission sync pulse to be received from the CC1200. This indicates that the first bit of the data is currently being modulated. The microcontroller then starts its own internal counter.
4. The radio modulates the ping data into radio waves and transmits it over the air.

5. The transponder's CC1200 receives the ping signal and detects the sync word. As soon as the sync word is decoded, its sync word detector outputs a signal internal to the chip.
6. After detecting sync, the transponder's CC1200 immediately switches into transmit mode (its auto-reply feature is enabled) and begins modulating a response. The time that this process takes is highly constant.
7. The ground station's radio receives the response's sync word and activates its sync detection output.
8. The ground station microcontroller accurately records the time when the sync pulse was received.
9. Later, the ground station microcontroller reads the packet from the radio and ensures that its contents are valid. If they are, the measurement is saved to flash memory. If not, the measurement is discarded.
10. The ground station begins waiting until the next time for a measurement.

Though it is fairly complicated, this process can occur in less than a millisecond! With four ground stations running, each ground station can be taking measurements at over 200Hz without interfering with each other. If all goes well, the ground stations take tens of thousands of range measurements over the course of the rocket's flight, which can then be averaged together to create significantly more accurate data. However, a more complete analysis than just averaging can also be performed.

Analyzing the Data

After the launch vehicle flies, and data is collected from each ground station, we move on to the next problem: determining the vehicle's actual flight path. This is more difficult than simply solving a single equation or computing an average, but is still computable using the right algorithm.

Each set of rangefinder measurements can essentially be seen as its own math problem: what point in 3D space has a distance to each ground station that matches what was measured, within experimental error? This type of problem is called trilateration and is, thankfully, well-documented. As long as at least three ranges from three different locations are available (and the vehicle is known to be above-ground), there is a single solution that gives the vehicle's location.

However, finding this solution is made more difficult by error in the measurements – the ranges from the ground stations will not agree on exactly one point, and might even disagree by hundreds of meters. To handle this complication, a more advanced algorithm is needed. In a study done by Navidi et al, several different methods of processing three-dimensional range data were evaluated, and it was found that a Non-Linear Least Squares (NLLS) estimator had by far the most accuracy in evaluating real-world range data compared to other, simpler solvers [4]. For processing our range data, we implemented a similar non-linear solver algorithm,

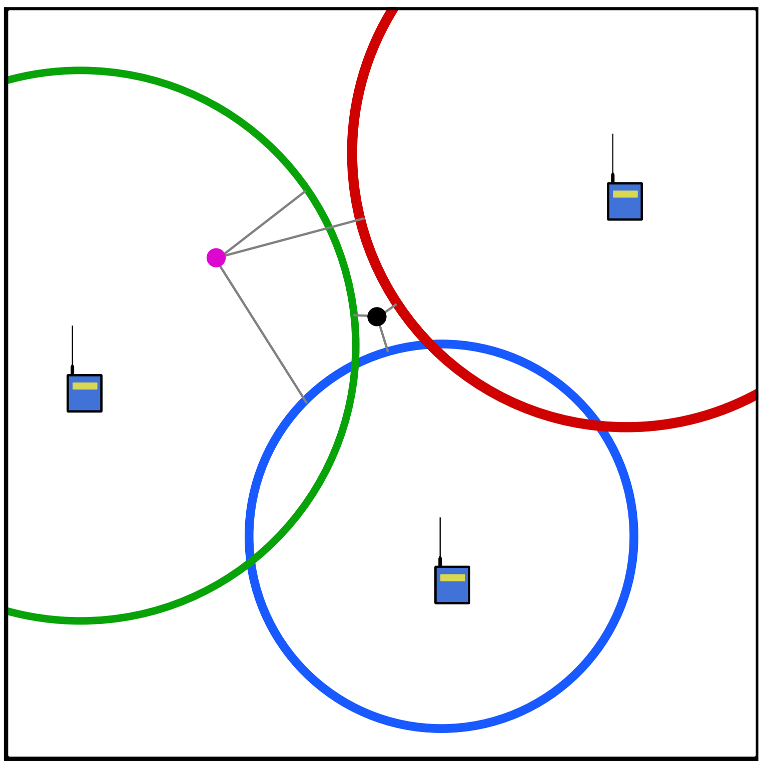


Figure 2.3. A 2D trilateration problem.

An example of the trilateration process, simplified to 2D for explanation's sake, can be seen in Figure 5. Each circle represents a range measurement, and measurement error has caused them to not overlap at a single point. One point the algorithm might consider might be the pink dot, but this has relatively high error to each measurement (error is represented by the sum of the squared lengths of the gray lines). The black point, in contrast, has very low error, so it would be very close to what the algorithm would finally select.

To solve rangefinder data, one must simply locate the point in 3D space with the smallest total error to each of the range measurements. This can be done with an iterative algorithm. First, an error function must be defined to quantify how "good" a point is. For a list $\overrightarrow{g_1}, \overrightarrow{g_2} \ldots \overrightarrow{g_i}$ of *n* ground station positions and a list $r_1, r_2 \ldots r_i$ of *n* ranges, the error function for point $\vec{p}$ is given by:

$$E(\vec{p}) = \sum_{i=1}^{n} (r_i - \|\overrightarrow{g_i} - \vec{p}\|)^2$$

Essentially, $E(\vec{p})$ is the sum of squares of the error between the distance the point *is*, and the distance it *should be* for each ground station. As the current point's distance gets closer to the range measurements from all of the ground stations, the error function approaches zero. This error function can then be fed into an iterative non-linear system solver such as the Levenberg-Marquardt Algorithm, which uses the error function's partial derivative to search for its minimum value. This global minimum of error is the algorithm's best estimate of position [4].

We implemented this algorithm and found it to be reasonably effective, especially for determining relatively vertical position. However, we are still looking into alternate methods that might provide an improvement for finding more accurate three-dimensional trajectories. Our results with this method are described in more detail in Chapter 5. But in the next chapter, we're going back to the beginning – to the very first conceptual designs of the rangefinder itself.

Chapter 3: The Development Process

Choosing a Radio

Development work on the rangefinder began in Fall 2019. For the very first step in what would become a storied development campaign, we needed to choose the basic design for our radio system for sending pings back and forth. Several different designs were considered, including analog circuits that simply transmit sine waves as well as digital modules like Texas Instruments (TI)'s CC2500 radio transceiver. We decided to use a digitally modulated radio to send data, as this method would provide much better immunity to signal echoes and interference than an analog radio system. In the end, we chose the TI CC1200 integrated circuit for a couple of reasons. First of all, it demonstrated high receive sensitivity, reaching better than -110dBm in its lower bitrate configurations. It also operates on the 430MHz and 900MHz bands, which are much more manageable for rocketry applications than higher bands like 2.4GHz – receiving these bands effectively requires a large and directional dish antenna, not just a simple omnidirectional wire. Last but not least, it was known that a fellow lab, the MIT Rocket Team, had successfully integrated a CC1200 onto their custom avionics unit. So, we knew that custom circuits with the CC1200 were possible – we just had to create one.

RangefinderTest

The next step in the process was to get some CC1200s on a board and begin testing with them. The decision was made to skip working with dev boards and go directly to manufacturing a custom PCB. This was partly due to the extortionate cost of CC1200 dev kits, but also because Rocket Propulsion Lab was secure in its well-practiced PCB building process, and building a custom PCB would ensure we could confidently work with CC1200s on future boards. However, the process of building the board, termed "RangefinderTest", turned out to be significantly tougher than expected.

The schematics were designed in late fall and winter 2019 for a board containing a microcontroller, a GPS (for time pulses), and two CC1200 radios. The circuits for the microcontroller (the NXP LPC1768) and the GPS (the U-Blox MAX8U) were based on preexisting RPL designs and were used in this board essentially unchanged. A new schematic was designed for the CC1200 radio circuit; this was essentially a conversion into Altium of TI's "CC120XEM_868_915" reference design for the CC1200 at 900MHz. Finally, the schematics were laid out into a PCB design (Figure 3.1) by the author (during several days on a family ski trip where this was by far the most interesting activity available).

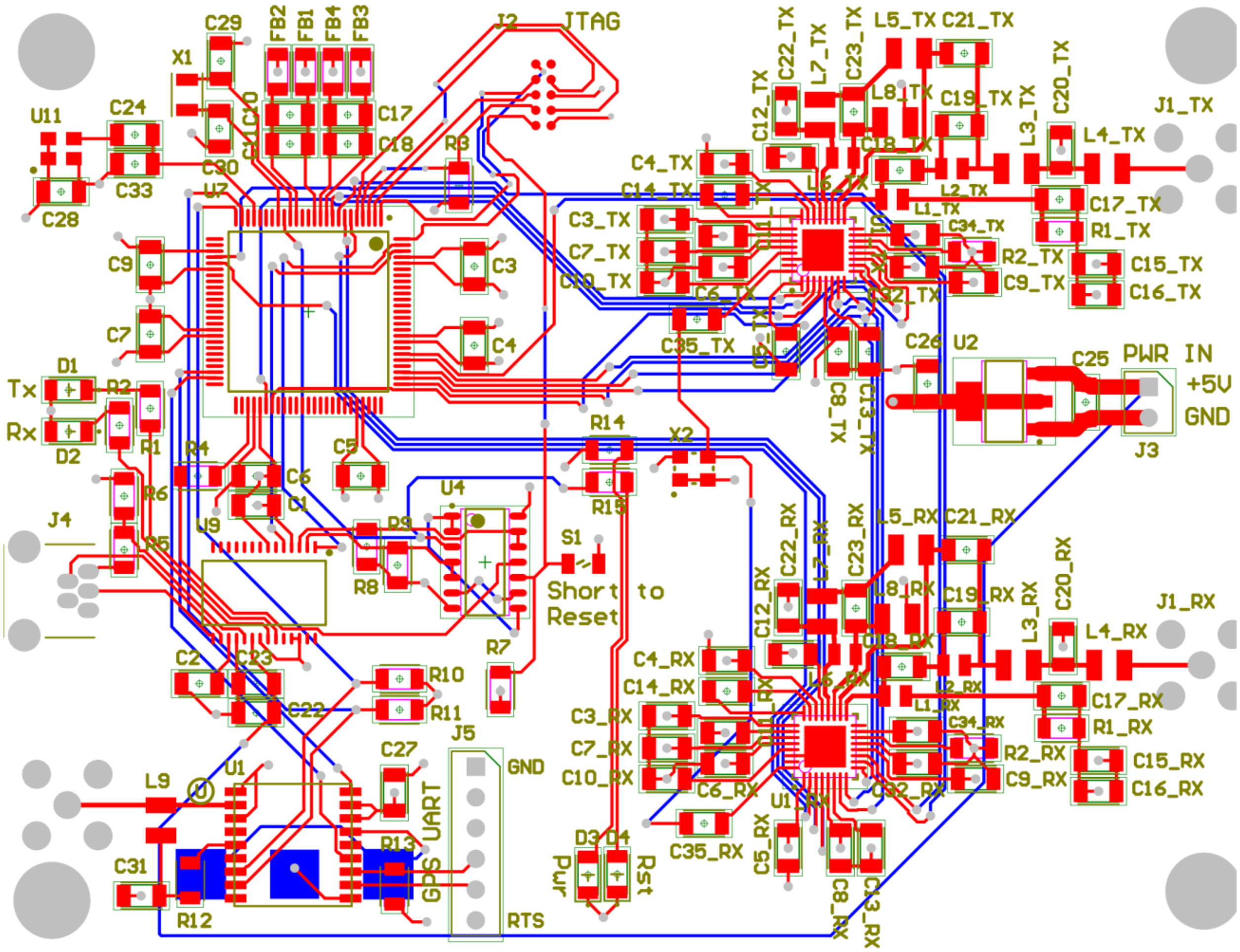


Figure 3.1. RangefinderTest PCB Layout

After initial routing, several more weeks were needed to design review the PCB with other lab members and incorporate changes, such as an easier-to-work-with JTAG connector. Finally, in late February, the PCB and its components were ordered. While we did not know it at the time, we were locked in a race against the oncoming COVID-19 pandemic. Just as the first reports of community transmission in the US were coming in, the board arrived on Rocket Propulsion Lab's delivery shelf. And the day before the university sent us home on a spring break that would last over a year, a skeleton crew of RPL members made the RangefinderTest board a reality (Figure 3.2).

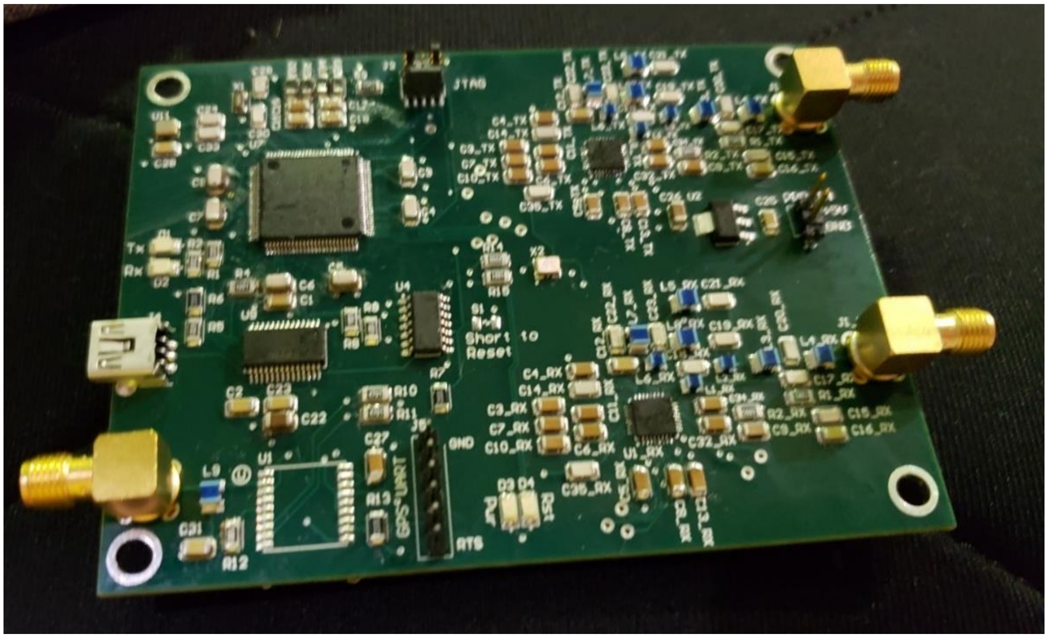


Figure 3.2. RangefinderTest board, as originally built.

Unfortunately, getting the board completely working proved to be more difficult than building it. When the pandemic hit in force, nearly all of RPL's electronics lab equipment was evacuated into the author's garage, and a temporary PCB rework setup was created (which turned out to be a lot less temporary than planned). Debugging of the board quickly revealed a major issue with its SPI bus: several traces that ran close together on the bottom of the PCB were shorted together in multiple places. While these traces were very close to each other, they were still further apart than the minimum allowed distance in the manufacturer's tolerances. As it turned out, many traces on the board, as well as some on another PCB that had been ordered at the same time, were shorted in this manner. After several weeks, consultation with our PCB manufacturer, Advanced Circuits, revealed that they had made a mistake on that batch of PCBs. They offered to replace the PCB, but we did not have the budget for a full set of replacement components to rebuild it, so we elected to repair it. The author proceeded to cut out all the sections of shorted traces and solder a network of fine-gauge rework wires to replace each cut trace (Figure 3.3). It wasn't pretty, but it worked.

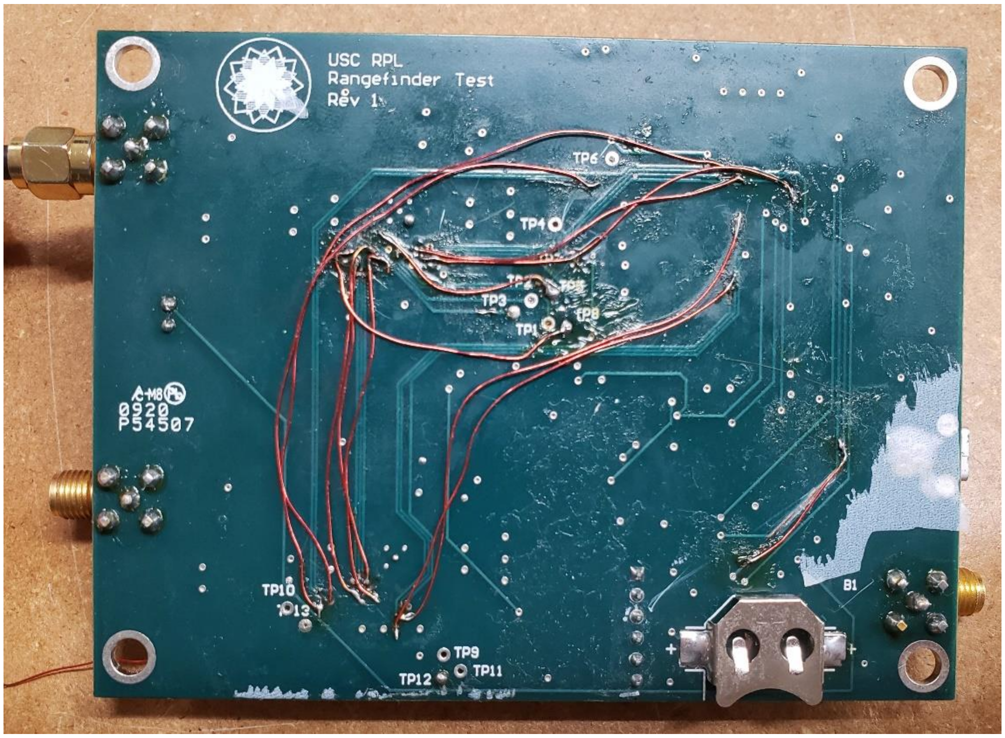


Figure 3.3. RangefinderTest board with traces repaired.

Once the traces were repaired, even more work was needed to make sure the CC1200s, fine-pitch 32-QFN integrated circuits, were getting soldered correctly. After several failed attempts to attach them, we finally discovered the key trick: applying strong downwards pressure on the center of the chip while soldering it made sure that it wasn't being lifted by solder accumulated on the center pad. Finally, after six attempts, a CC1200 was successfully soldered and was contacted by our software (Figure 3.4).

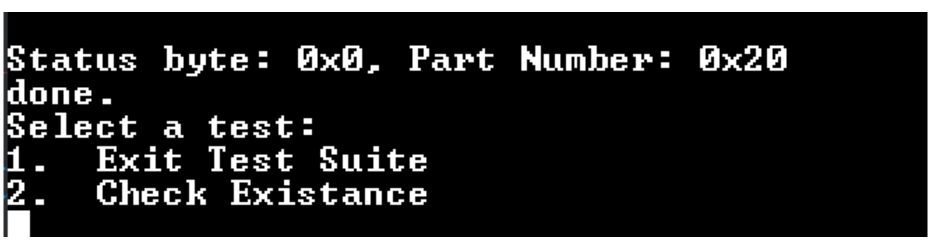


Figure 3.4. A small triumph: getting a response from the CC1200.

After all these fixes, the RangefinderTest board was finally working as designed. In short order, we wrote the code for an early version of the jitter test (see Test 1 in Chapter 4) and began testing.

Timing Design

Over the next few months, we went through a painful process of learning what features of the radio were and weren't dependable for accurate timing. For instance, we discovered that the CC1200's SPI bus interface was not deterministic at all – there could be several microseconds of jitter between asking it to begin transmitting data and the data actually getting sent, which was far too much for accurate ranging. This knocked out our original plan, which was to simply have the processor enqueue a message on the CC1200 and begin timing from when the SPI command is sent. What was deterministic, however, was the CC1200's PKT_SYNC_RXTX output. This GPIO pin pulses high when the first byte of a packet is sent or received [5, p. 18]. This output seems to be coupled to the radio part of the transceiver IC, and it seems to be accurate to within about +-175ns of when the packet is sent or received under optimal radio settings (see Test 1) (though averaging reduces this error significantly).

This led us to our current timing design: when the ground station sends a packet, a CPU timer input records the first pulse on PKT_SYNC_RXTX indicating that the packet has been transmitted. Then, once the packet is received on the transponder, the transponder uses the radio's auto reply feature (which we also determined to be deterministic) to send a response ping immediately without involvement from the processor. Finally, the ground station receives the response ping and the CPU records the second pulse on PKT_SYNC_RXTX marking when the response came in. By measuring the time between these two pulses, the distance from the transponder can be calculated. This process proved to work well and be as accurate as the system would allow. By the end of the summer, we were able to obtain results such as Test 1. This meant that it was time to move to the next phase of the system!

Radio Revisions

Based on testing with the RangefinderTest board, as well as further design reviews internal to RPL, we had identified a few things that needed fixing on the system before we could build the next revision. First and foremost were improvements to the link budget.

For those who don't know, link budgeting is essentially a process of adding up all of the radio transmit powers, losses, and receive sensitivities in a system, subtracting expected loss from the distance the radios are to be used at, and then determining the power margin at this distance. The power margin gives the strength of the signal at the target distance relative to the minimum strength needed to receive the signal. As long as the power margin is greater than zero, the radios can theoretically communicate, though this communication is likely to be very unreliable. In real systems it is recommended to have 5-10dBm of power margin to handle additional unanticipated losses.

Using the Friis Transmission Equation [6, p. 428], the power decrease, called the path loss, of a radio signal as it travels through the air can be calculated as $20\log_{10}(\frac{4\pi df}{c})$, where *c* is the speed of light, *d* is the distance the signal travels through the air (in m), and *f* is the frequency. Rearranging for *d*, we can also calculate that if the allowable loss in decibels of the radio link is *L*, the theoretical maximum distance that the radios can communicate at is $\frac{c\times 10^{\frac{L}{20}}}{4\pi f}$. Note that while frequency does not directly affect how a signal propagates, it affects how effectively an omnidirectional antenna can receive it, so we include it under the path loss term per convention. This provides a very useful way to make best-case estimations, as long as the signal really is propagating though free space only.

As Figure 3.5 shows, the link budget of the original system provides only a +2dBm power margin at 100km. Technically this is enough to reach our goal if all estimates are accurate, but for RPL's planned applications to high-power rockets, it was insufficient. Factors such as poor antenna orientation (hard to avoid on a free-spinning rocket), additional cable or antenna losses, or worse-than-predicted hardware performance could significantly change this. If, for example, 5dB were lost anywhere in the system, range would be reduced to 74km, below our requirements – and this level of loss is almost expected from a whip or dipole antenna on the rocket in a poor orientation.

Stage	RF Power	Notes
Transmit Power	+27dBm	Rated output power of CC1200 radio with CC1190 amplifier
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors

Ground Antenna Gain	+9dBi	Using Yagi antenna
Rocket Antenna Gain	0dBi	Assuming dipole antenna on rocket
Receiver Loss	-1dBm	Ballpark loss from cables and connectors
Receiver Sensitivity	- (-100dBm)	Worst-case sensitivity of CC1200 with low-noise amplifier
Path Loss	-131.5dBm	*f = 900MHz, d = 100km*
Total Margin	**+2.5dBm**	

Figure 3.5. Link budget of original radio design.

So, we investigated all of the methods available to us, and decided on two actions to take to buy some additional safety margin. First of all, we switched to the 430MHz radio band instead of 900MHz. Since path loss has a frequency term, decreasing frequency by about half bought us about 6dBm of additional budget. Second, we switched from TI's recommended CC1190 amplifier IC to Skyworks' bidirectional SKY65366 amplifier. This amplifier fulfills essentially the same role in the circuit but has 4dBm higher output power.

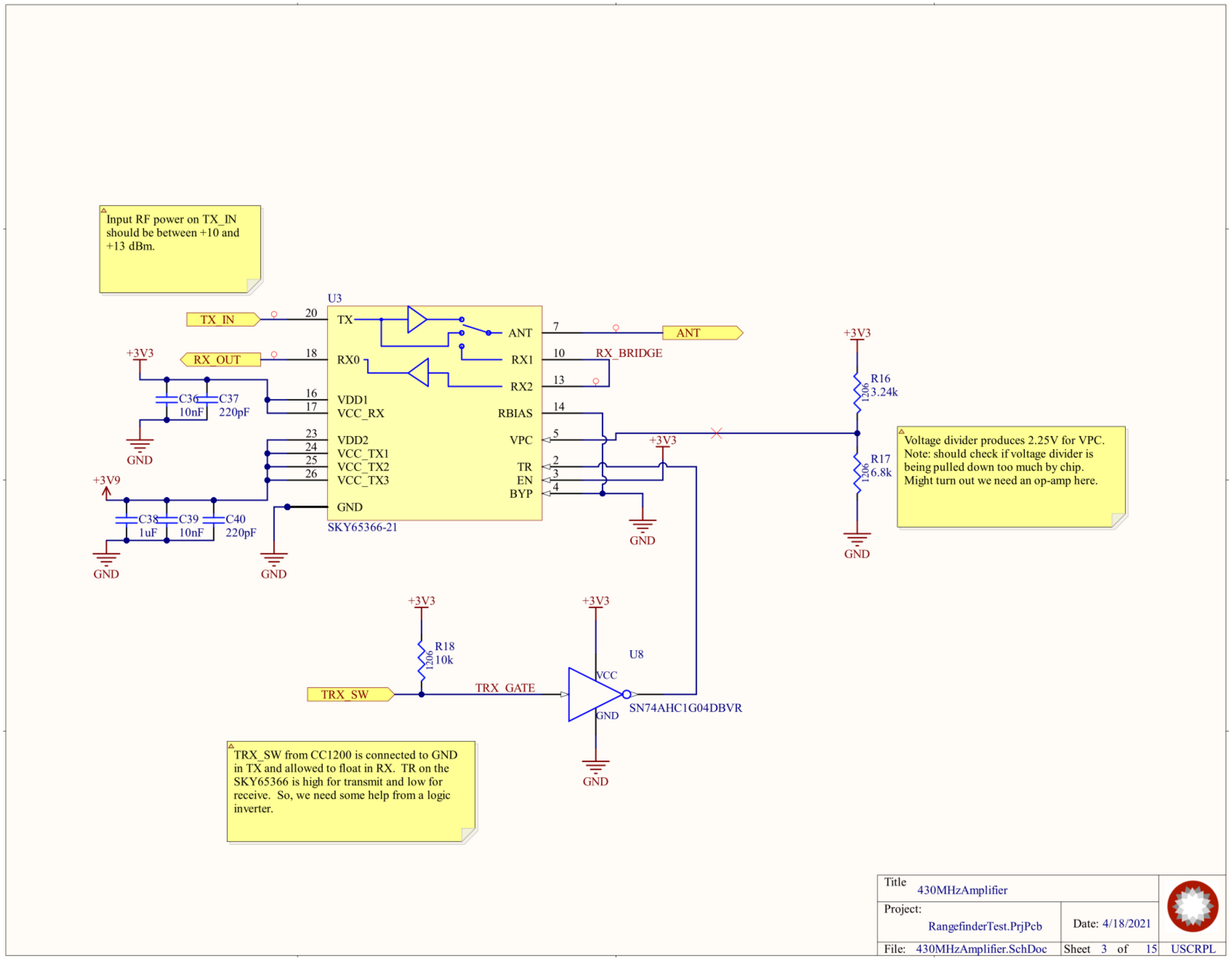
Stage	**RF Power**	**Notes**
Transmit Power	+31dBm	Rated output power of SKY65366
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors
Ground Antenna Gain	+9dBi	Using Yagi antenna
Rocket Antenna Gain	0dBi	Assuming omnidirectional antenna on rocket
Receiver Loss	-1dBm	Ballpark loss from cables and connectors

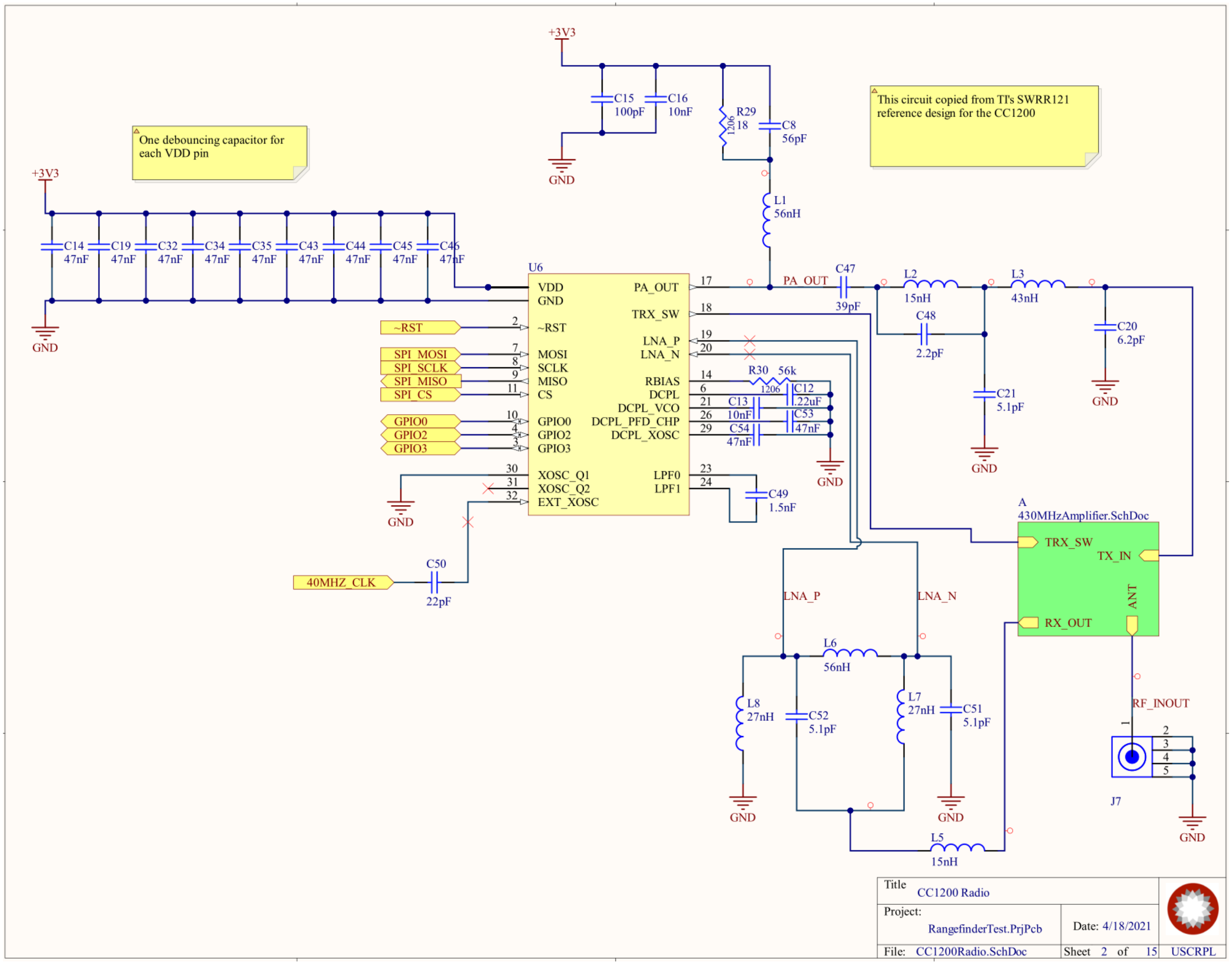
Receiver Sensitivity	- (-100dBm)	Worst-case sensitivity of CC1200 with low-noise amplifier
Path Loss	-125dBm	*f = 430MHz, d = 100km*
Total Margin	**+13dBm**	

Figure 3.6. Link budget of revised radio design.

Though these changes (Figure 3.6) are small, they add up to make a significant difference in the system's performance. We obtained a hefty +13dBm power margin at 100km, which should be enough to compensate for poor antenna orientations and other weirdness. By setting the power margin to zero and solving for distance, we can also determine the system's theoretical maximum range as 441km, so we expect to be covered even if we significantly exceed our previous apogee. (Note that path loss is highly nonlinear, so a small increase in total power margin produces a big increase in distance).

The final schematics of the radio system we designed are given on the next two pages.


Input RF power on TX_IN should be between +10 and +13 dBm.
U3
TX_IN
RX_OUT
TX
RX0
VDD1
VCC_RX
VDD2
VCC_TX1
VCC_TX2
VCC_TX3
GND
ANT
RX1
RX2
RBIAS
VPC
TR
EN
BYP
SKY65366-21
RX_BRIDGE
+3V3
+3V9
C36 10nF
C37 220pF
C38 1uF
C39 10nF
C40 220pF
GND
R16 3.24k
R17 6.8k
R18 10k
1206
Voltage divider produces 2.25V for VPC. Note: should check if voltage divider is being pulled down too much by chip. Might turn out we need an op-amp here.
TRX_SW
TRX_GATE
U8
VCC
SN74AHC1G04DBVR
TRX_SW from CC1200 is connected to GND in TX and allowed to float in RX. TR on the SKY65366 is high for transmit and low for receive. So, we need some help from a logic inverter.
Title 430MHzAmplifier
Project: RangefinderTest.PrjPcb
Date: 4/18/2021
File: 430MHzAmplifier.SchDoc
Sheet 3 of 15
USCRPL


+3V3
C15
100pF
C16
10nF
R29
18
C8
56pF
GND
One debouncing capacitor for each VDD pin
This circuit copied from TI's SWRR121 reference design for the CC1200
L1
56nH
C14 C19 C32 C34 C35 C43 C44 C45 C46
47nF
U6
VDD
GND
~RST
MOSI
SCLK
MISO
CS
GPIO0
GPIO2
GPIO3
XOSC_Q1
XOSC_Q2
EXT_XOSC
PA_OUT
TRX_SW
LNA_P
LNA_N
RBIAS
DCPL
DCPL_VCO
DCPL_PFD_CHP
DCPL_XOSC
LPF0
LPF1
~RST
SPI_MOSI
SPI_SCLK
SPI_MISO
SPI_CS
GPIO0
GPIO2
GPIO3
PA_OUT
C47
39pF
L2
15nH
L3
43nH
C48
2.2pF
C20
6.2pF
C21
5.1pF
R30 56k
C12
.22uF
C13
10nF
C53
47nF
C54
47nF
C49
1.5nF
40MHZ_CLK
C50
22pF
A
430MHzAmplifier.SchDoc
TRX_SW
TX_IN
ANT
RX_OUT
RF_INOUT
J7
LNA_P
LNA_N
L6
56nH
L8
27nH
C52
5.1pF
L7
27nH
C51
5.1pF
L5
15nH
Title CC1200 Radio
Project: RangefinderTest.PrjPcb
Date: 4/18/2021
File: CC1200Radio.SchDoc
Sheet 2 of 15
USCRPL

The Ground Station

With this improved radio design in hand, it was time to construct the rangefinder ground stations. Besides the radio system, GPS, and CPU that we had designed, each ground station needed user interface elements, such as a display and buttons. It would also need memory for storing data, for which we added both an SD card slot and a 64MiB flash memory IC (the same module as used on RPL's flight avionics unit). Finally, since wall power wasn't available, each ground station would need its own 3.3V power regulator, as well as a 3.9V regulator to power the SKY65366 RF amplifier. For power in the field, we elected to add an XT60 connector for standard hobby lithium batteries because they were ubiquitous, cheap, and stored a substantial amount of energy. Over two months were spent designing and reviewing these schematics, and finally, a PCB was routed from them (Figure 3.7).

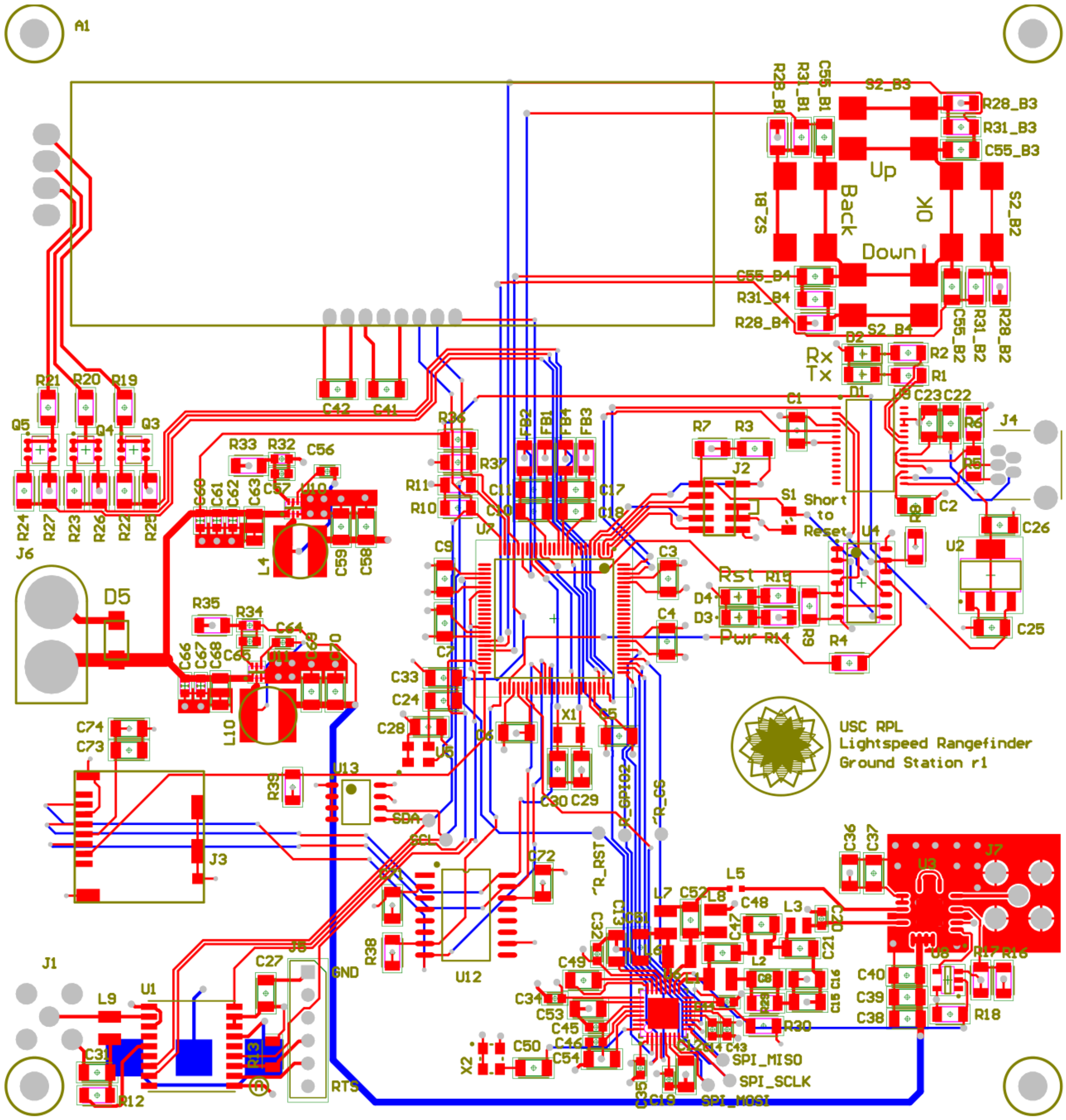


Figure 3.7. PCB layout of the ground station.

While four ground stations were needed for a complete system, Rocket Propulsion Lab elected to order only two initially, wanting to make sure that the fundamental design worked before pouring a lot of resources into building it. The first PCB (Figure 3.8) was hand-built by the author, still working out of the temporary garage space, and was turned on with high hopes. However, these were dashed by a sudden eruption of flame from the voltage regulator section of the board when the battery was plugged in. The voltage regulators had fried themselves, and taken nearly every other chip on the board with them.

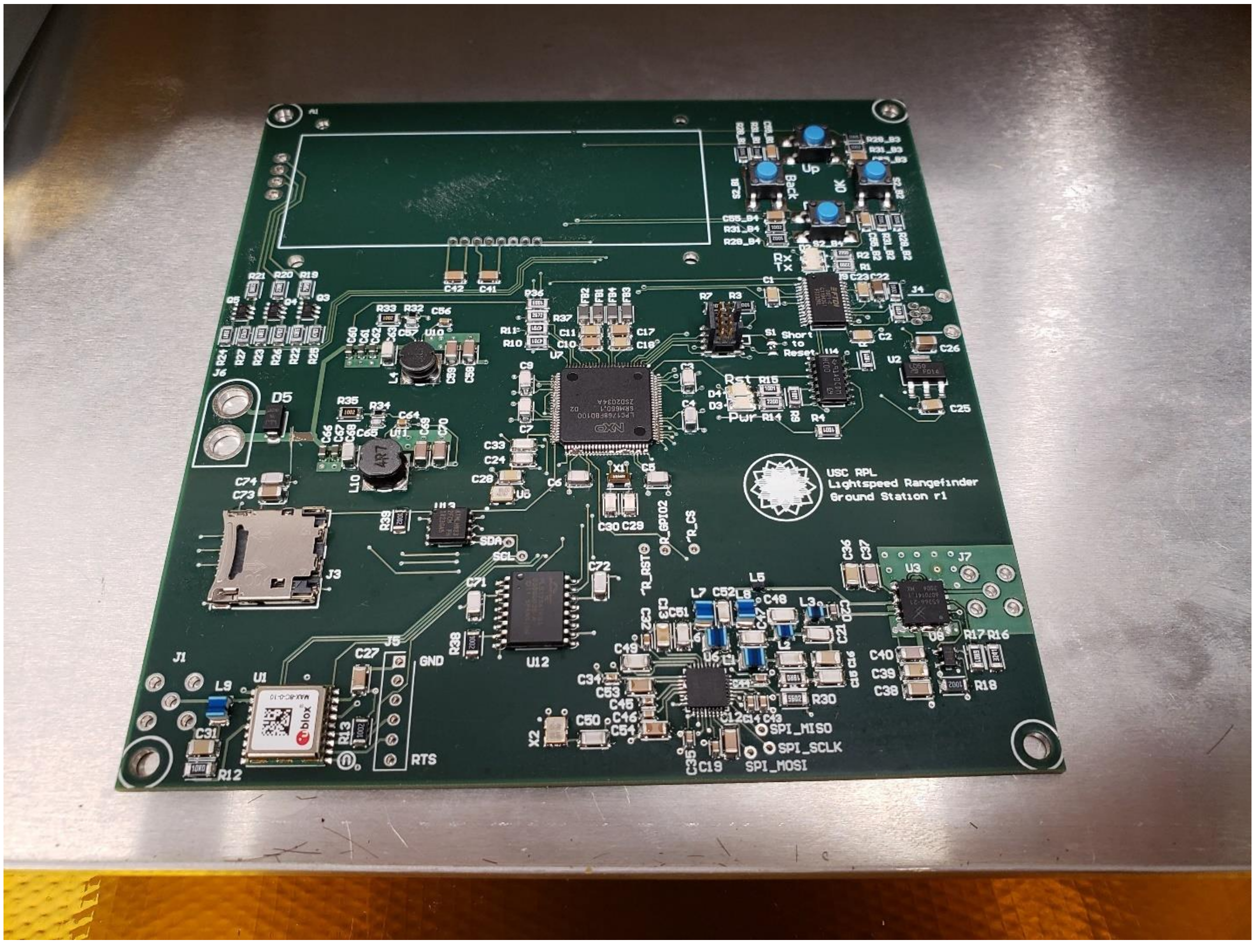


Figure 3.8. Ground station board, fresh from the reflow oven.

Further testing revealed the source of the problem. Previous RPL boards had only been used with benchtop power supplies or low-capacity batteries. These power sources have relatively high internal resistance, meaning that when the boards were first plugged in to power, the input capacitors on the voltage regulators charged up slowly. In contrast, the hobby lithium batteries were designed for motor control and had much lower internal resistance (meaning that they could supply a lot more power when needed). This meant that there was a huge current spike when the batteries were first plugged in to the board, and due to the inductance of the battery wires, this current spike caused a significant voltage spike when current flow ceased. The following trace (Figure 3.9) was taken of the VBAT rail on the board after plugging in power.

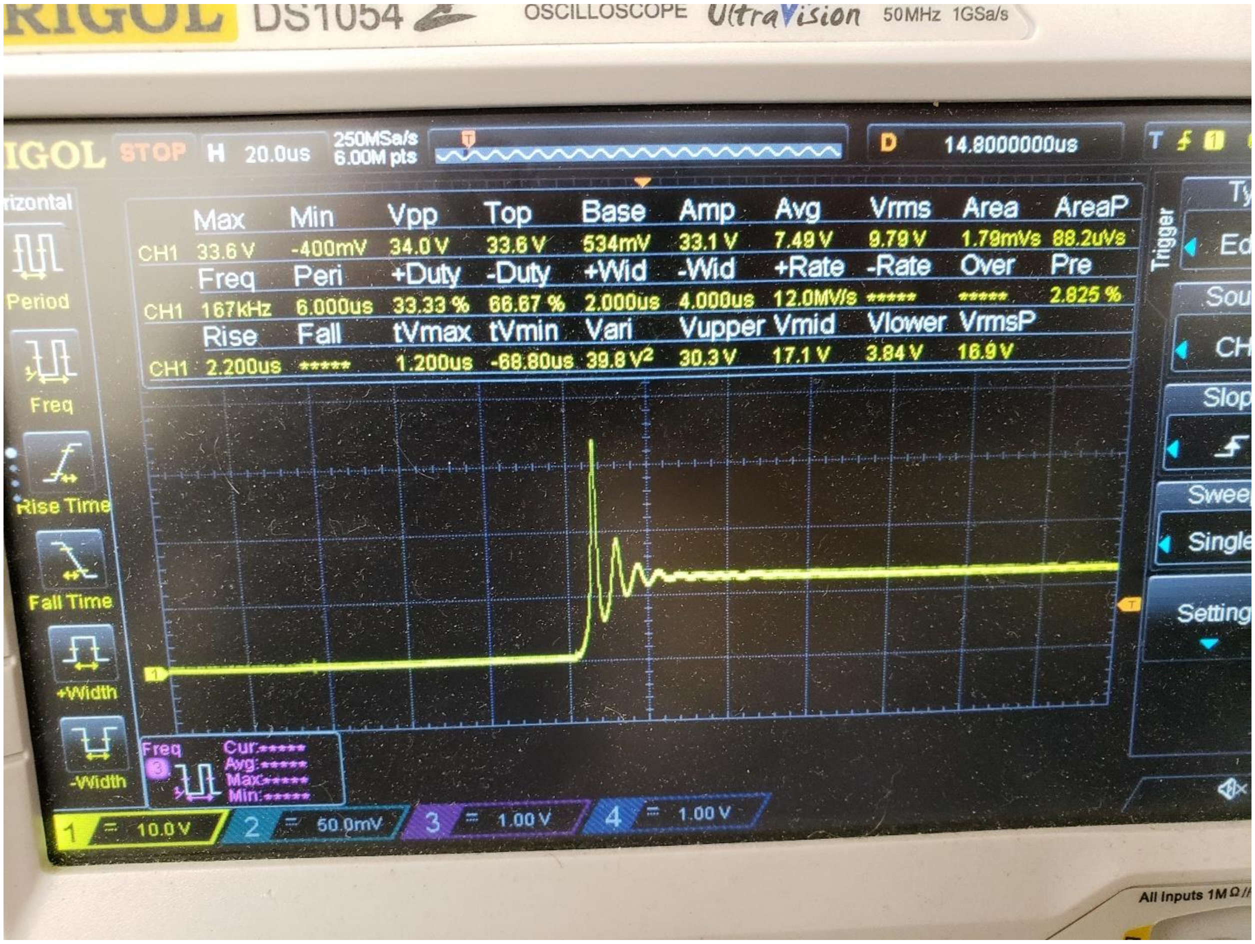


Figure 3.9. Voltage spike on power on.

Shockingly, the spike reached 33 volts! The TPS562207 regulators that we chose for the board had an absolute maximum input voltage of only 19V [7]. (I say "had" because after the voltage spike they became tiny charred lumps of plastic and silicon, not TPS562207s). Evidently, this voltage spike was damaging their internal power transistors and forcing them into being permanently on, causing the regulators to internally short battery voltage to ground and then explode soon after from the dissipated power.

In the short term, remedying the problem was easy once we understood what was going on. We simply added a 1Ω power resistor in series with the battery power input (Figure 3.10). Additionally, nearly every digital IC on the board had to be replaced with a spare, as the failing regulators had sent battery voltage into the regulated digital power rails. After these actions, a much cleaner power-on trace was observed (Figure 3.11), and the board finally began to work.

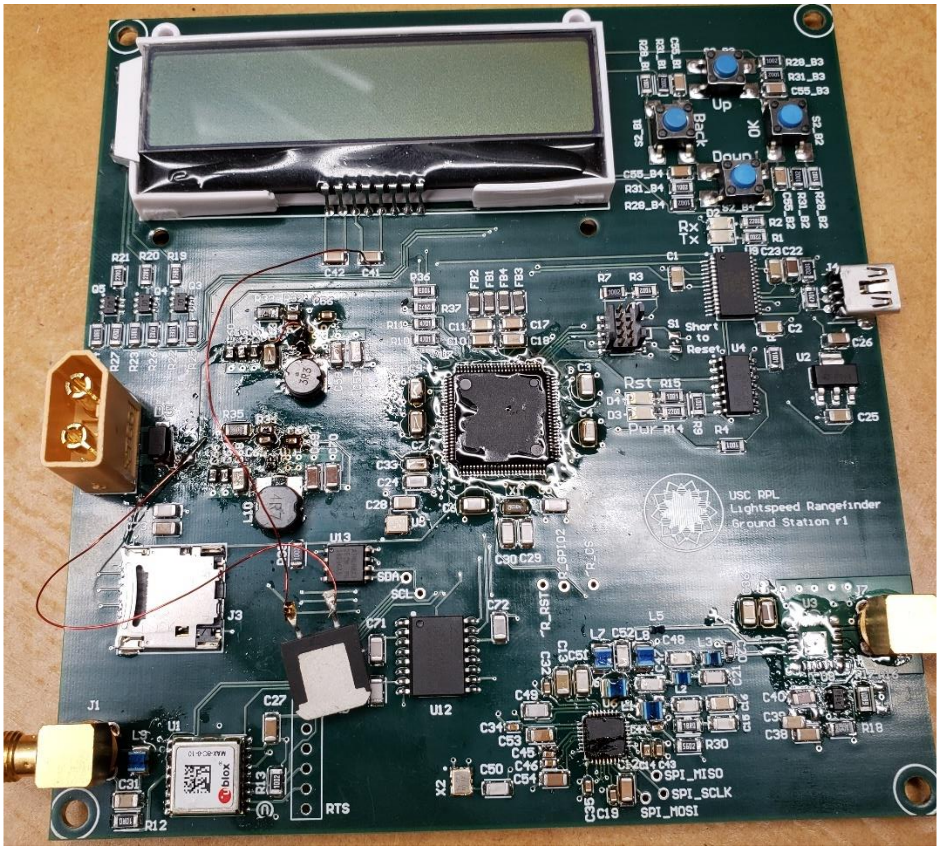


Figure 3.10. Jumpering on a power resistor.

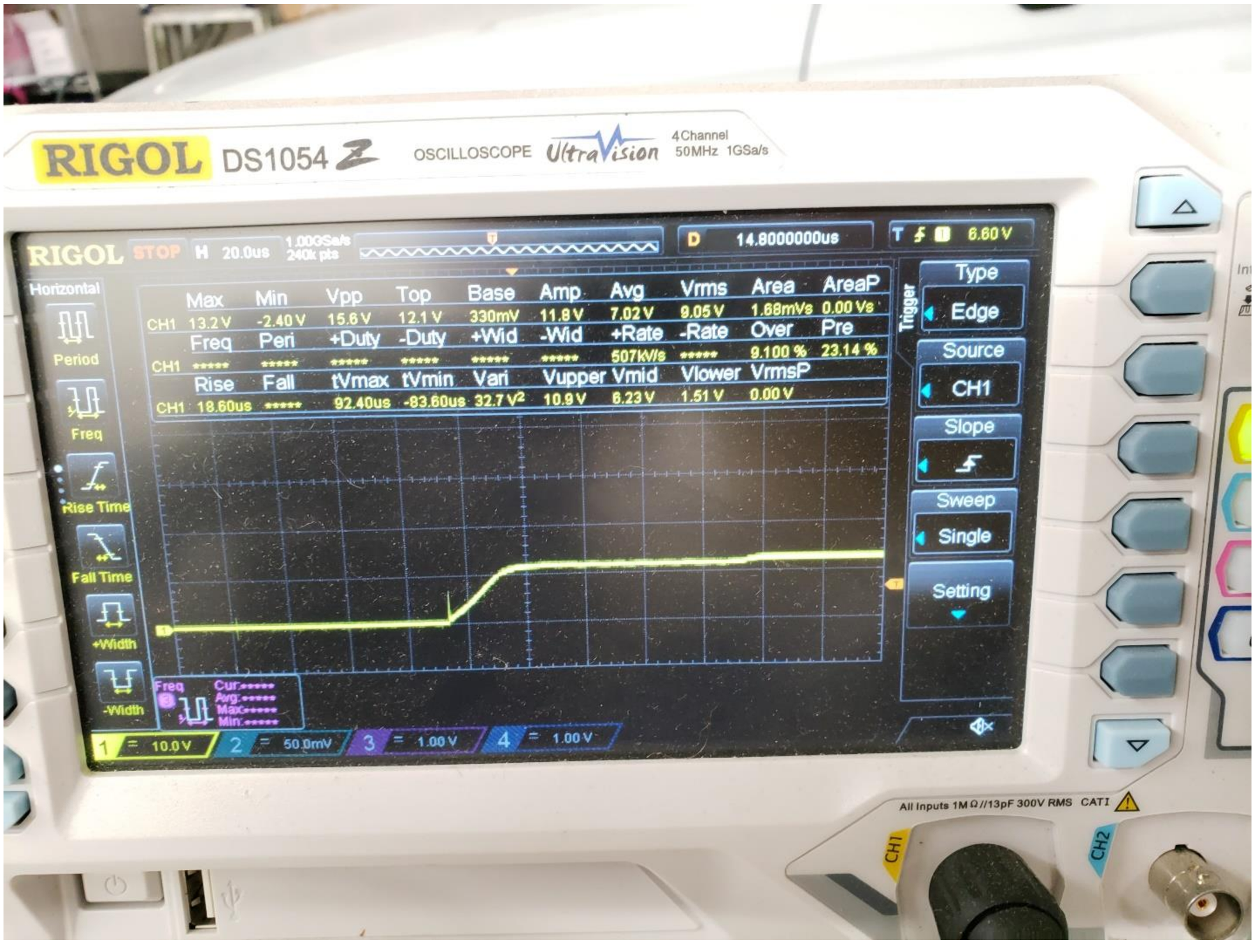


Figure 3.11. A much cleaner power-on trace!

Work then proceeded on the board's firmware. Code was found from Mbed OS community sources, as well as RPL's existing codebase, to control the flash memory, EEPROM, display, and SD card on the board. For accurate timekeeping, an extended version of RPL's U-Blox GPS driver was written with timepulse support, and timing code was written to use an interrupt to align the CPU's internal clock with GPS seconds. This provides a microsecond-accurate timebase for the ground stations to synchronize their transmissions with each other. Several bugs in RPL's CC1200 chip driver were found and fixed, and finally the radios were successfully tested by sending a transmission from one rangefinder to the other. Initially RF output power was anomalously low, but this was remedied by removing and resoldering the SKY65366 amplifier IC (Figure 3.12). This resulted in a much nicer RF power output level (Figure 3.13).

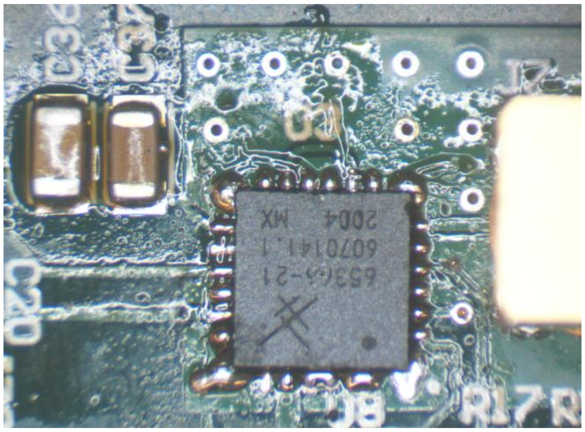


Figure 3.12. SKY65366 after being resoldered (note that visible solder bridges are between pins that are supposed to be connected). Microscope view.

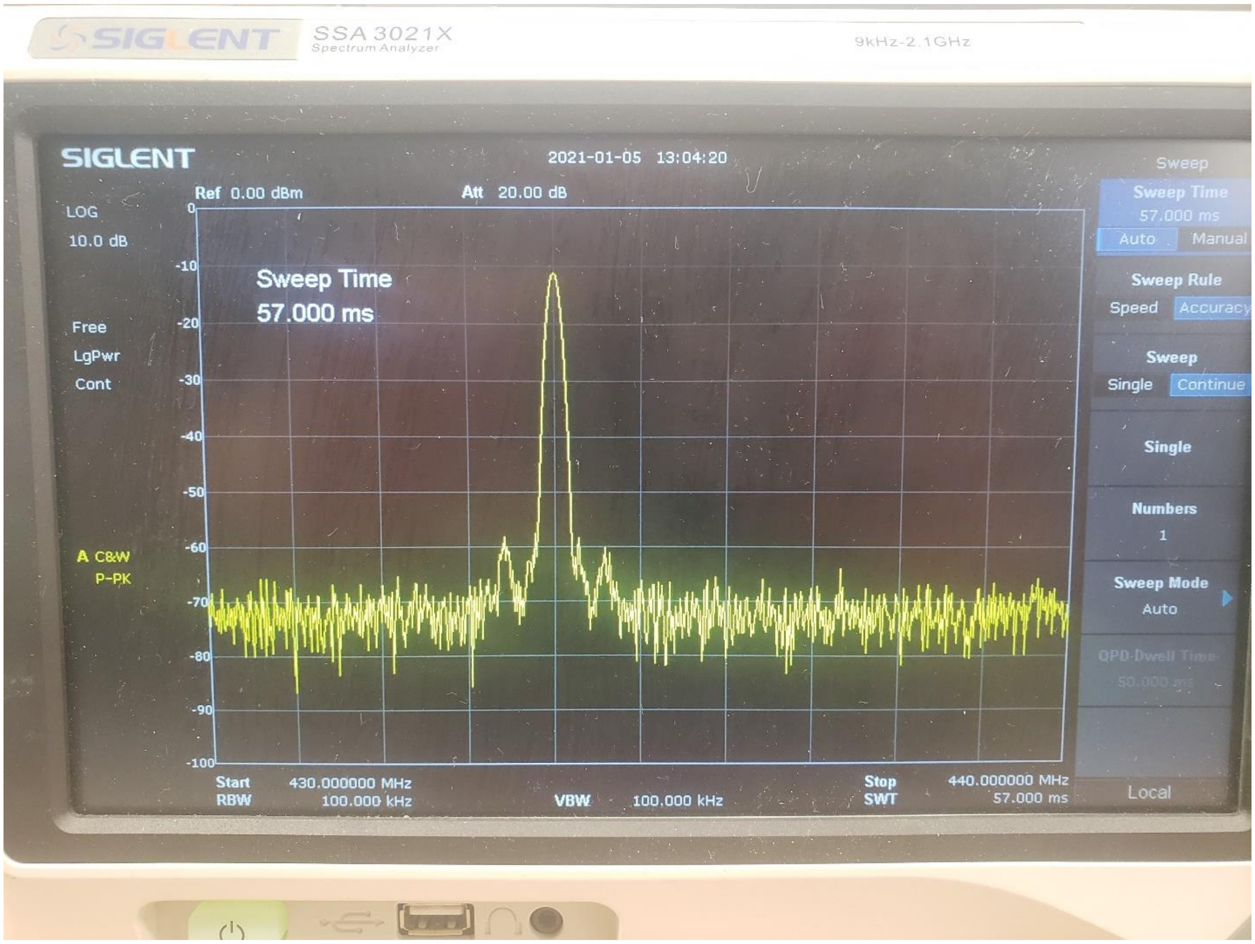


Figure 3.13. Ground station board transmitting an unmodulated sine wave. ~40dBm of attenuation has been applied.

With all external devices working, the main loop of each ground station could then be written and tested. This was implemented as a state machine (Figure 3.14), and handles the process of transmitting the amateur radio call sign (as Morse code), acquiring GPS lock, and sending ranging pings at the appropriate times.

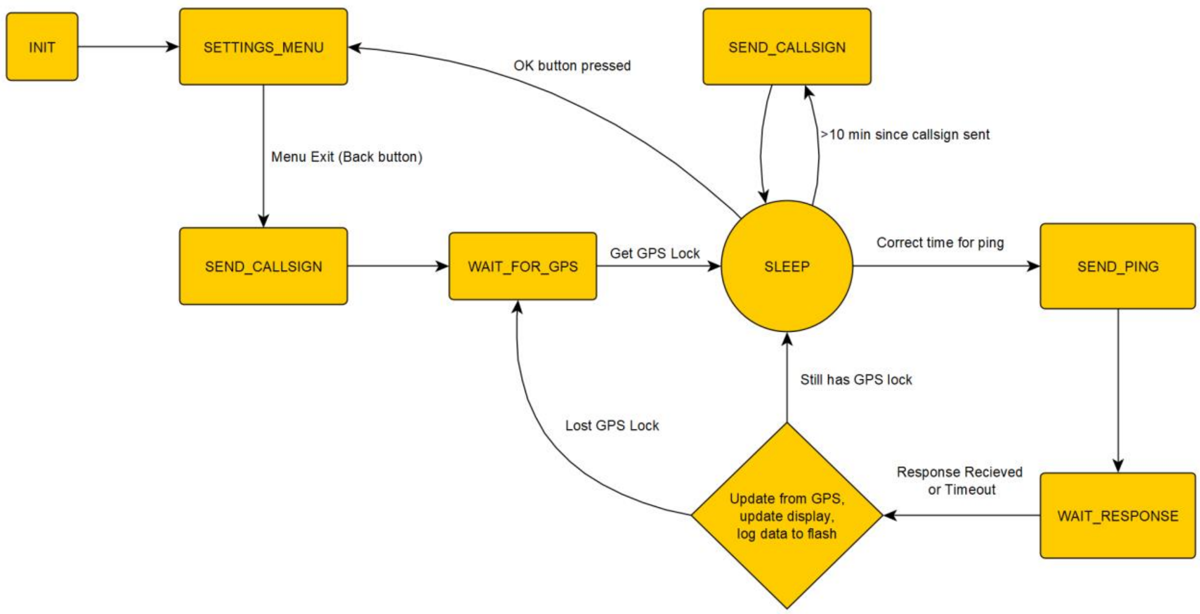


Figure 3.14. State machine of each ground station

Testing activities continued for several more weeks, culminating in some successful ranging experiments (see Test 5). However, limited physical range was observed, and issues with the receiver side of the rangefinder were confirmed using the process from Test 3. Both ground stations were displaying very low receive sensitivity, in the vicinity of -65dBm when they should have been about -100dBm. This issue proved difficult to debug due to the large number of moving parts in the system: the test setup, the PCB, the RF amp, the CC1200 IC, and the software settings all played a role in the effective sensitivity. First, we changed the easiest thing: the software. We replaced our custom radio settings, optimized for minimum jitter, with the default ones. This had a measurable impact, but only improved the sensitivity to -78dBm, not the expected -100dBm. While there was clearly a software and configuration problem, there had to be other issues at hand.

Importantly, both PCBs had displayed the exact same results in the sensitivity testing. So, we concluded that the issue was unlikely to be an assembly issue or component failure, and was instead some manner of design problem.   To narrow it down, we removed the SKY65366 amplifier from the PCB and bridged the receive input directly to the SMA jack (Figure 3.15). After removing the amplifier from the circuit, the sensitivity improved to -86dBm. But this was a bit of a conundrum. How could a low-noise amplifier actually have been *weakening* the radio signal? We double-checked all the schematics and electricals, wondering if there might've been some connection for the amplifier that we set up incorrectly. We even experimented with cutting and rerouting some of the traces for the amplifier based on an alternate interpretation of the datasheet. (Pro tip: if you ever get a wild hair and want to remove a QFN pad and replace it with a jumper wire, don't. It's a lot more trouble than it's worth.) In the end, however, it turns out that we were just unlucky: there was in fact a soldering problem in the receive pathway, and it just so happened that both boards had the exact same problem. After several attempts at resoldering the SKY65366, both units were operating at their full -100dBm receive sensitivity at last (see Test 3).

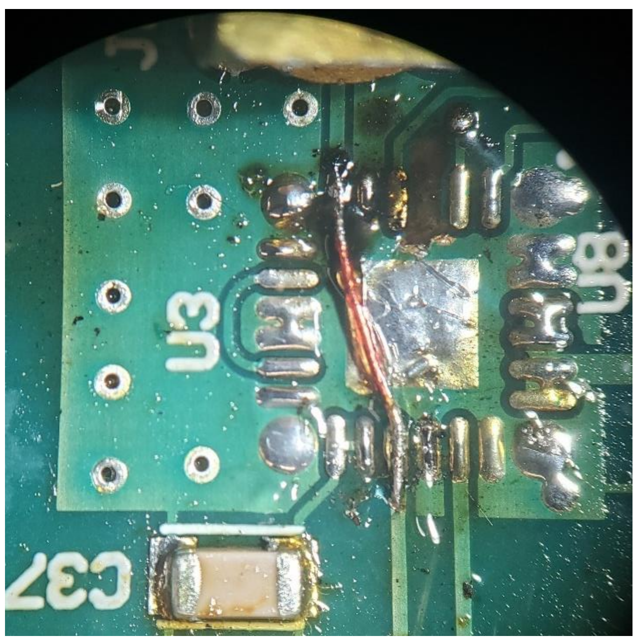


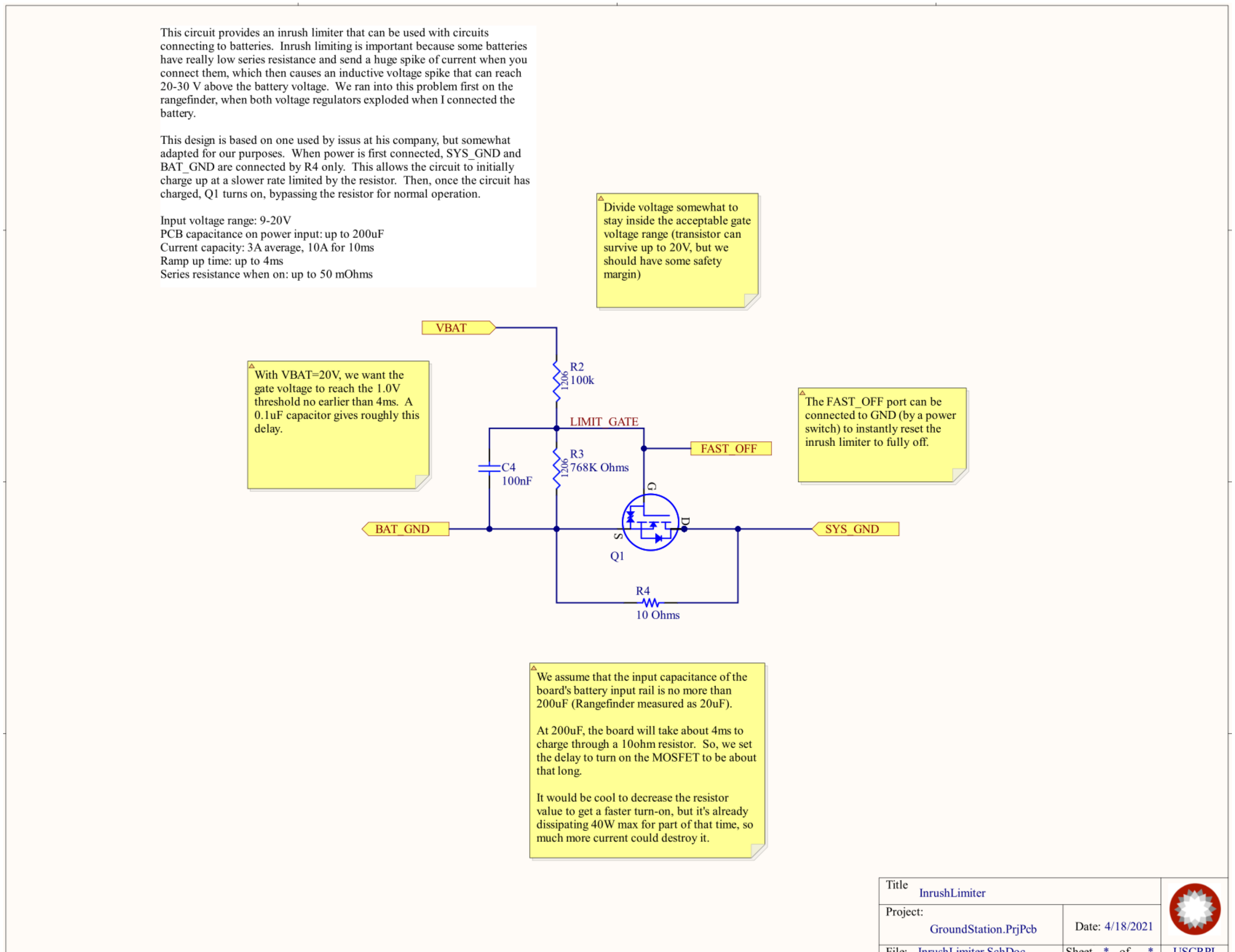
Figure 3.15. Bridging the SMA jack to RX_OUT. Microscope view.

The Transponder and the Shortage

With all aspects of the radio system tested, we were finally ready to build the transponder and additional ground stations needed for a complete rangefinder system. There was just one, small, insignificant, *massively devastating* problem: the integrated circuit shortage of 2021 had now hit in full force. In particular, two chips were now completely unobtainable: the LPC1768 (RPL's venerable CPU) and the SKY65366 amplifier. No matter how hard we looked across the internet, we couldn't find a reliable source of these two chips. We would need to design a new version of the rangefinder without them.

We were unable to locate a direct replacement for the SKY65366. This chip contains a transmit power amplifier (PA), a receive low-noise amplifier (LNA), filtering, and an RF switch in a single package. This was a massive convenience for us, as many of the more difficult to design and tune components were predesigned for us. Losing access to it was a massive blow, and made the job of creating the rangefinder significantly harder. Luckily, we did have some old designs we could draw on: RPL had previously used the CML CMX902 amplifier IC for another 430MHz band project. This chip provided impressive transmit power (33dBm) in a small package. However, it also had a number of drawbacks: it required complicated external impedance matching and power supply circuits, it was poorly documented, and it only contained a transmit amplifier (PA), not a receive amplifier (LNA). However, it was the best option we could find at the time, so we used it. In order to match the functionality of the SKY65366, however, we needed to pair the CMX902 with a RF switch (an IC which connects the antenna to either the receive or the transmit side of the CC1200). For this job, we selected the PE4259, a popular, high-power, and easy to integrate RF switch. Based on sensitivity information from testing the RangefinderTest board and from the CC1200 datasheet, we decided not to add an LNA, since the CC1200 should have been able to reach a sensitivity of -99dBm or so without assistance. With these two ICs, and supporting components, we had all the pieces of a radio front-end – the next question was whether it would work.

Before we could order the board, there was one other issue that needed addressing: the exploding voltage regulators from months prior. Putting a power resistor in series with the battery input slowed the charging enough to solve the problem, but was very wasteful of energy since power would be dissipated in the resistor as long as the board was running. After consulting with Mark Harris, a professional electrical engineer who had dealt with similar issues, we created the following schematic, the InrushLimiter. It works by, at first, first having the battery and board connected through a high-power resistor (R4), just like the original fix. However, the RC circuit acts as a timer which activates Q1 after a few milliseconds, providing a low-resistance path around R4. This provides the best of both worlds, offering a soft start while not wasting any power while the circuit is operating.


This circuit provides an inrush limiter that can be used with circuits connecting to batteries. Inrush limiting is important because some batteries have really low series resistance and send a huge spike of current when you connect them, which then causes an inductive voltage spike that can reach 20-30 V above the battery voltage. We ran into this problem first on the rangefinder, when both voltage regulators exploded when I connected the battery.
This design is based on one used by issus at his company, but somewhat adapted for our purposes. When power is first connected, SYS_GND and BAT_GND are connected by R4 only. This allows the circuit to initially charge up at a slower rate limited by the resistor. Then, once the circuit has charged, Q1 turns on, bypassing the resistor for normal operation.
Input voltage range: 9-20V
PCB capacitance on power input: up to 200uF
Current capacity: 3A average, 10A for 10ms
Ramp up time: up to 4ms
Series resistance when on: up to 50 mOhms
Divide voltage somewhat to stay inside the acceptable gate voltage range (transistor can survive up to 20V, but we should have some safety margin)
VBAT
R2
100k
1206
With VBAT=20V, we want the gate voltage to reach the 1.0V threshold no earlier than 4ms. A 0.1uF capacitor gives roughly this delay.
LIMIT_GATE
FAST_OFF
The FAST_OFF port can be connected to GND (by a power switch) to instantly reset the inrush limiter to fully off.
R3
768K Ohms
1206
C4
100nF
G
D
S
BAT_GND
SYS_GND
Q1
R4
10 Ohms
We assume that the input capacitance of the board's battery input rail is no more than 200uF (Rangefinder measured as 20uF).
At 200uF, the board will take about 4ms to charge through a 10ohm resistor. So, we set the delay to turn on the MOSFET to be about that long.
It would be cool to decrease the resistor value to get a faster turn-on, but it's already dissipating 40W max for part of that time, so much more current could destroy it.
Title InrushLimiter
Project: GroundStation.PrjPcb
Date: 4/18/2021
File: InrushLimiter.SchDoc
Sheet * of *
USCRPL

We also needed to replace the now-unobtainable CPU. After a trade study of a couple of different possibilities, RPL selected STMicro's STM32L452RE for the rangefinder, as well as for future "small" jobs around the avionics unit that require a dedicated processor. This IC is small, inexpensive, uses little power, and requires few external components – in total, it uses less than half the board area as the LPC1768 did on LPC1768-based designs. Despite this, it has almost the same processing power as the LPC1768, as well as other useful features like asynchronous communications bus support and high-current IO pins. The exact design of the CPU circuit is outside the scope of this paper, but we stuck fairly closely to the chip's reference design.

Finally, all of the pieces were together, and it was time to create the transponder itself. The transponder was meant to be a small, standalone rangefinder unit that simply responded to the ground stations' pings. This meant it didn't need the GPS subsystem, the user interface, or any nonvolatile storage. We also wanted it to be as small as possible to maximize the amount of amateur rockets that it could fit inside – many rockets are only a few inches in diameter, and we didn't want size to be the reason that we couldn't fly. So, we set out to create the smallest board possible containing the CPU, power circuitry, and radio from the full-size rangefinder (Figure 3.16). In retrospect, the focus on small size may have made life more difficult (we had to use large quantities of 0603 components), but it came out as a beautiful-looking PCB (Figure 3.17).

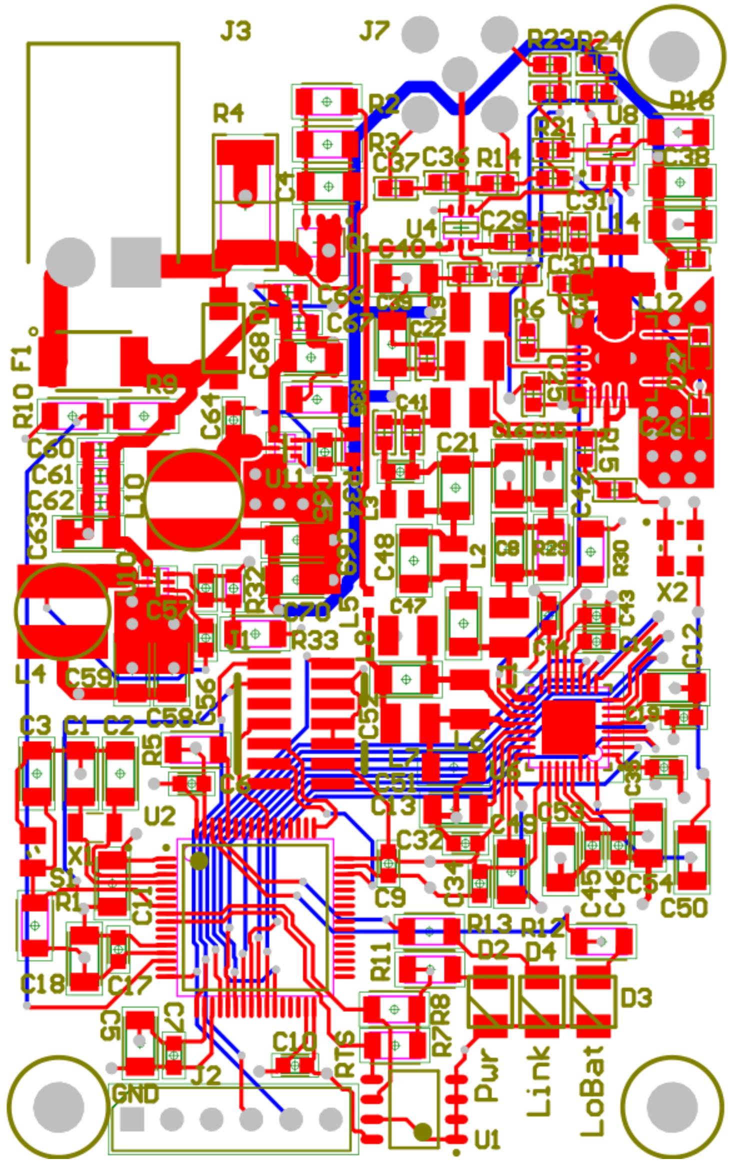


Figure 3.16. Transponder PCB layout

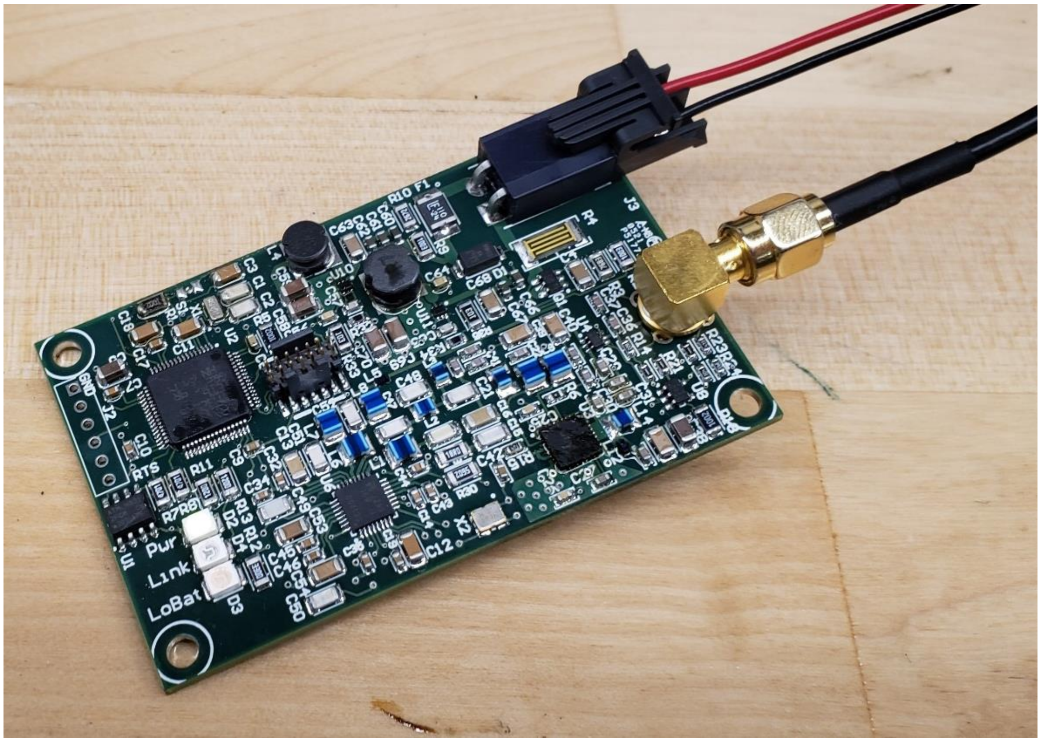


Figure 3.17. Assembled transponder PCB

Unfortunately, not all of the newly designed circuitry on the transponder worked out of the box. In particular, it struggled with receive sensitivity. Despite our earlier projections that the radio would work sufficiently well without an LNA, it was only able to receive at -81.5 dBm. Several possible explanations for this behavior were considered. It is a possibility that the CC1200 simply requires an LNA in order to reach -100dBm receive sensitivity on 430MHz. We had seen this performance on the RangefinderTest board operating at 900MHz, but that was a different frequency band so the behavior might've been different. However, another theory is that the poor sensitivity is due to sub-optimal routing of the receive radio trace on the PCB. As is visible in Figure 3.18, this trace is over an inch long between the RF switch and the CC1200 radio. This could cause significantly more attenuation for received signals than the short trace on the ground station. However, attenuation for this trace (a controlled impedance microstrip line about an inch long) shouldn't have been more than 2dBm, so by itself this is not enough to explain the performance drop by almost 20dBm.

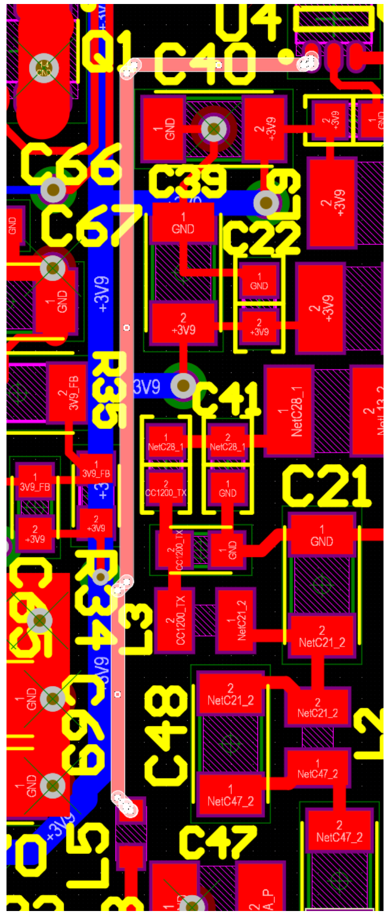


Figure 3.18. RF receive trace highlighted.

The best theory put forward is that the issue was due to noise coupling with the power regulators. Due to the push to miniaturize the board, the 3.9V regulator powering the amplifier was put less than 0.3” to the left of the RX trace, and a number of other voltage regulation components are even closer to this trace. While we cannot detect it with the equipment we have, it is likely that noise from the voltage regulator coupled onto the RF receive trace. This voltage regulator outputs a square wave at a low frequency (500kHz), but (as we confirmed via a SPICE simulation) this has many harmonics that stretch into the gigahertz. The amplitude of these harmonics is low, but the noise amplitude would only need to be tens of microvolts to exceed the amplitude of a -90dBm radio signal, preventing the automatic gain control (AGC) of the CC1200’s receiver from increasing the gain and detecting the weaker signal. Without more sensitive equipment we cannot quantify this behavior, but it provides a plausible explanation for why the receive performance of this board is so much worse than all other boards we’ve created.

The transponder's CMX902 TX amplifier also demonstrated some strange behavior. Initially, it worked, but at lower power than expected – while it should have been able to reach +33dBm maximum output power, the most we were able to reach was +27dBm, and even that required putting in a much stronger input signal than should have been required. Furthermore, at that output level, the amplifier generated a ridiculous amount of heat – it resembled a miniature supernova, and could trigger its own thermal shutdown if it was left on continuously. We suspected a soldering problem, as issues with the CMX902 footprint on the PCB made soldering the chip difficult. Several attempts at resoldering and replacing the chip did not change its behavior, but we believed that close tolerances on the PCB might be causing the output of the amplifier or another line to become shorted to ground, causing the heating.

Due to high heat output, high power consumption, and unpredictable behavior seen in tests (see Test 4), we eventually elected to remove the CMX902 from the transponder PCB entirely, and directly bridge the CC1200's transmit output to the antenna with a jumper wire. This caused heavy losses since the jumper wire was not an impedance controlled trace; output at the SMA jack was only +11dBm instead of the +14dBm that the CC1200 was rated to produce. However, this seemed to be the only way to get the board working reliably, so it was a sacrifice we had to make. Thankfully, with this change, the transponder finally began to operate reliably (albeit at reduced link budget). So, it was time to create the final two ground stations!

The Ground Station V2

Since two more ground stations were still needed for the full system, it was time to roll up all that we'd learned from the first ground stations and the transponder into an improved version of the ground station: the Ground Station V2.

First, we attacked the radio amplifier issues seen on the transponder. The PCB footprint for the CMX902 was enlarged and spaced out in the hopes of making the chip easier to solder. Additionally, a 10dBm inline attenuator was added between the output of the CC1200 and the input of the CMX902. This reduced the signal amplitude enough that the CC1200 could access the full output range of the power amplifier (previously, the lowest possible amplitude from the CC1200 was still one of the highest permissible input amplitudes for the CMX902). This assuaged worries that the CC1200 could have been overdriving the CMX902 and damaging it.

We also attempted to remedy the poor receive sensitivity of the transponder by adding a low-noise amplifier to the receive path. We were still unsure if it was necessary, as there had been other possible explanations for the transponder's poor receive performance. But with the flight test coming up, we wanted to maximize our chances of a successful connection. For simplicity, we went with the PSA4-5043+, the same chip used in the ubiquitous LNA4ALL product. We integrated it into our design using a public schematic [8], and placed it as physically close to the RF input as was feasible. Hopefully, this would amplify the received signal before it had a chance to weaken as it moved across the board.

A few other changes were also made besides the radio fixes. Since the LPC1768 was still unavailable, we replaced it with L452RE processor from the transponder, with a more accurate temperature-compensated oscillator and a totally redesigned timer scheme to work with the L452RE's more limited hardware timers. Also, we integrated the inrush limiter circuit, which had worked on the transponder with no issues. Finally, some quality-of-life improvements were added, such as a power switch, bright LEDs for feedback to the operator, and a battery fuse in case of a short. With that, we routed the PCB (Figure 3.19), sent if off, and hoped for a low-drama build (Figure 3.20).

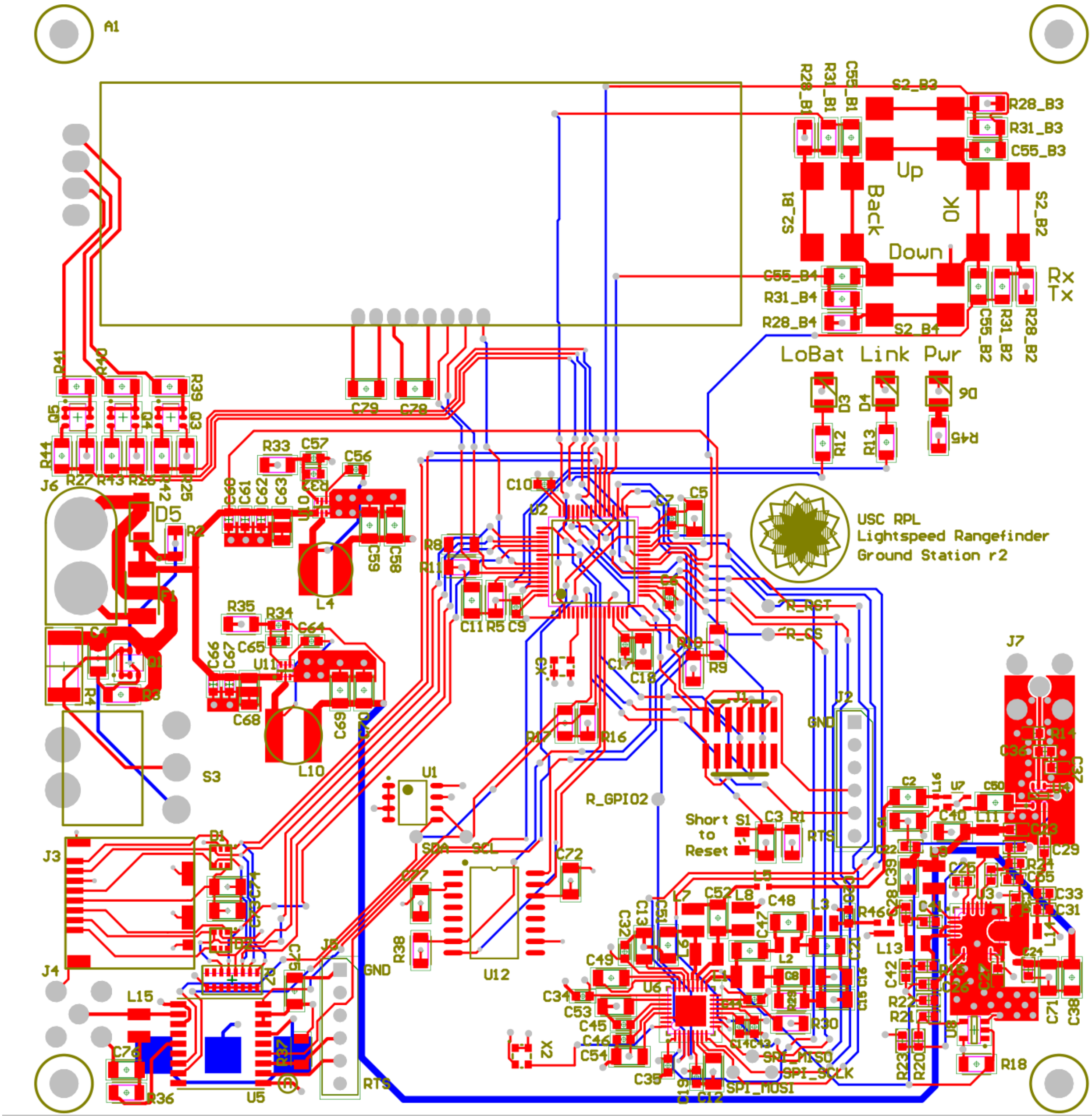


Figure 3.19. Ground Station V2 PCB layout

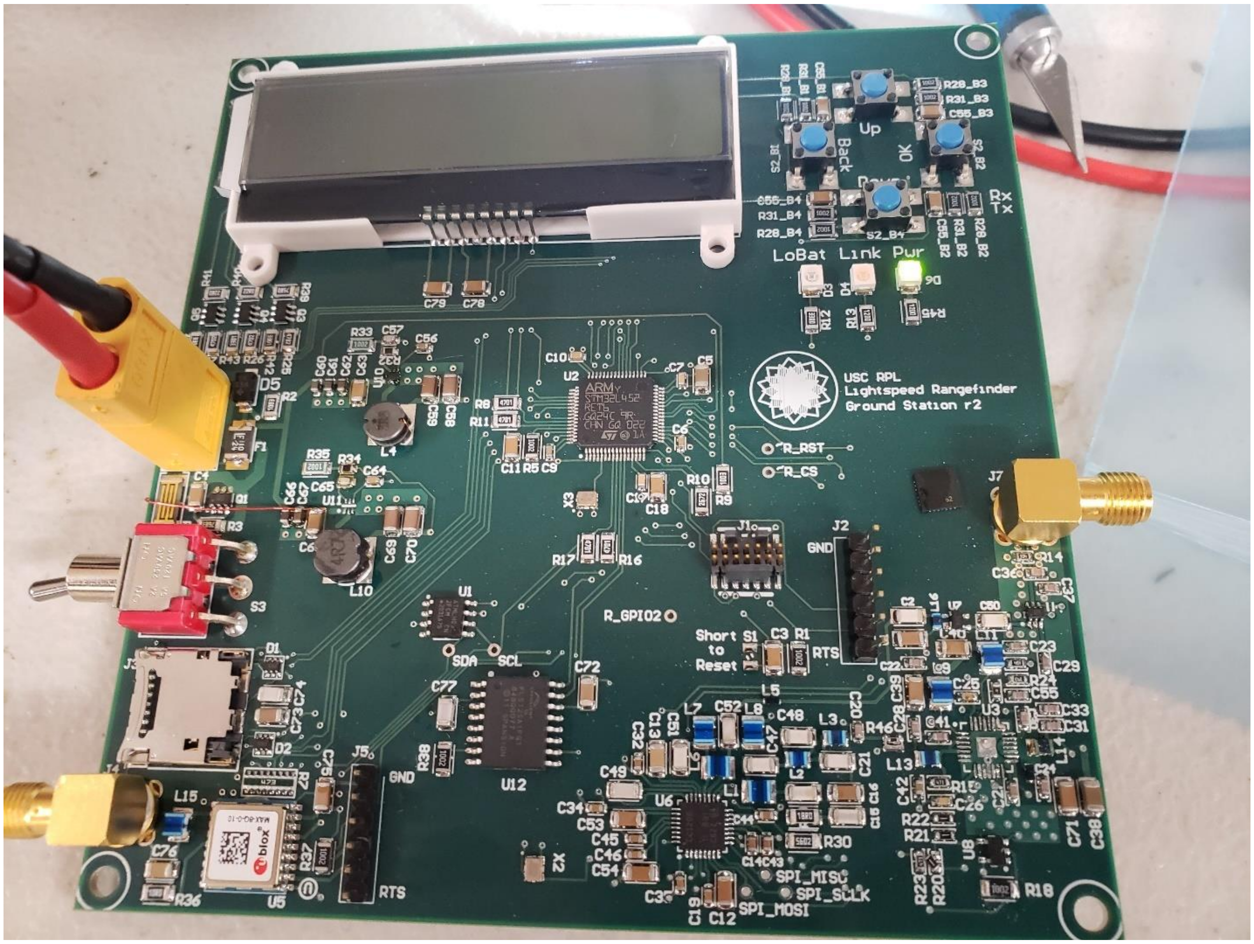
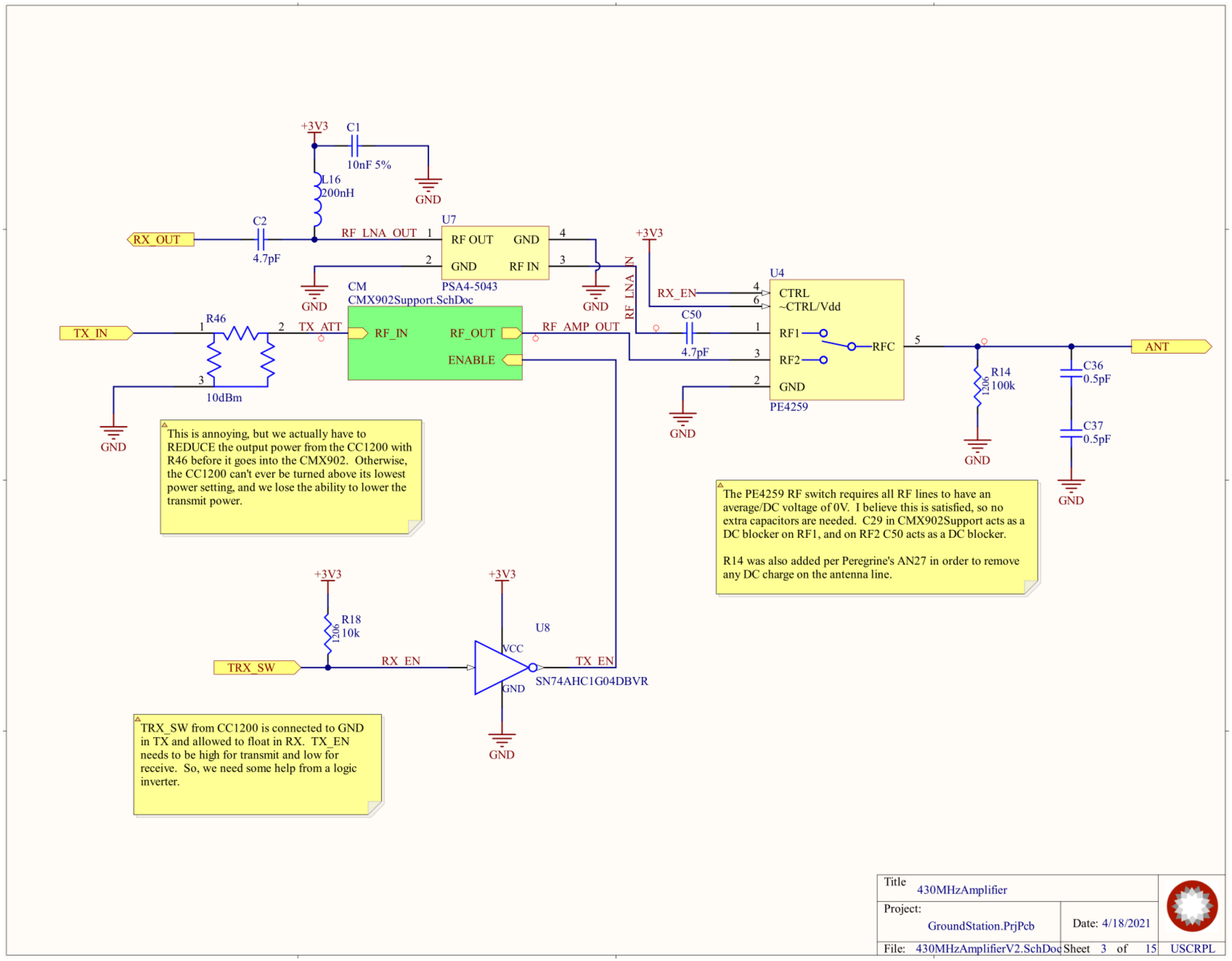


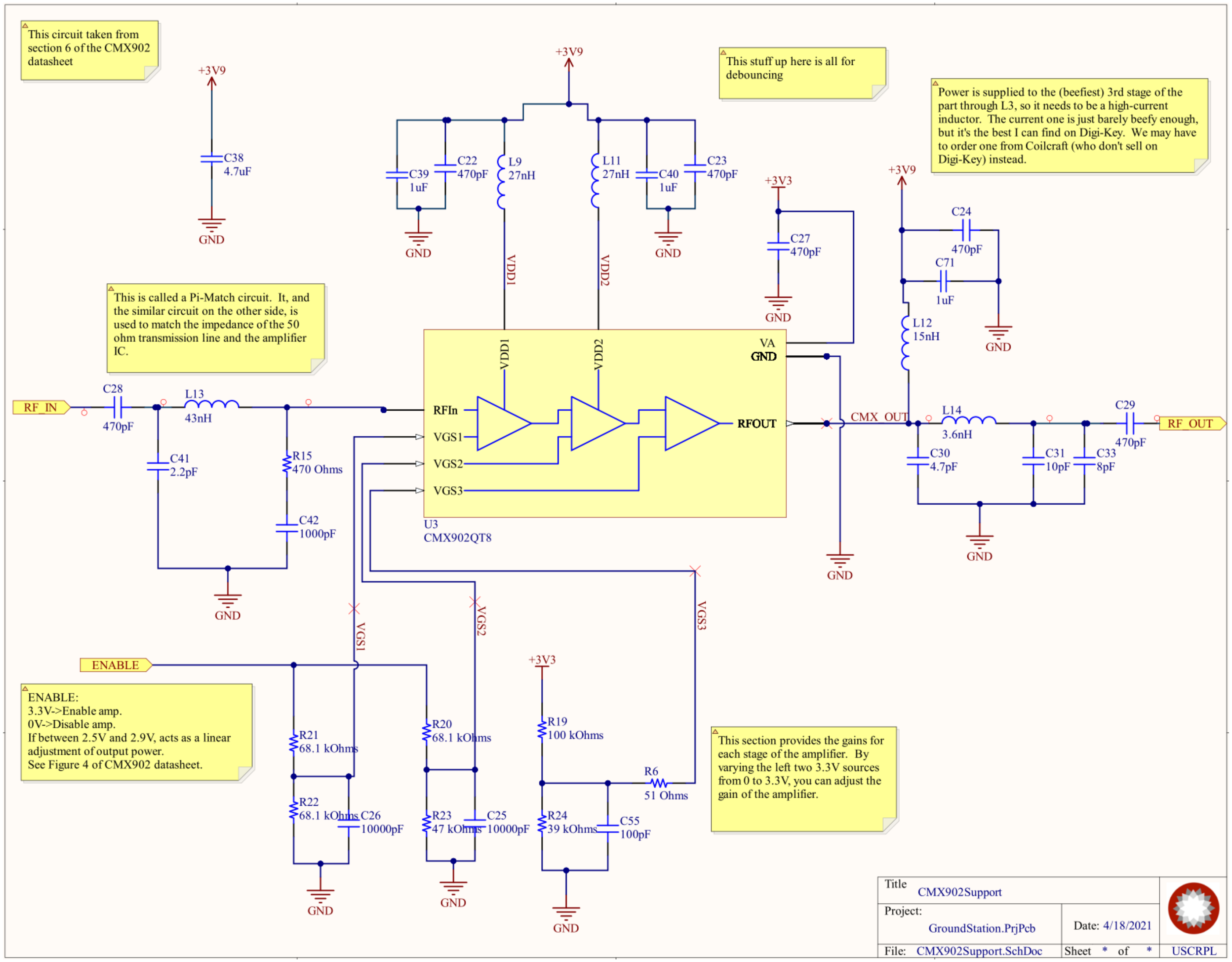
Figure 3.20. Completed Ground Station V2 PCB. CMX902 was temporarily removed at time of photo.

The goal of a "low-drama" build was… not totally achieved. In a horrible echo of four months prior, the board's 3.9V regulator exploded as soon as the battery was plugged in (at least I was wearing safety glasses this time). This time around, we were never able to find the cause. We confirmed with an oscilloscope that the inrush limiter was doing its job and preventing voltage spikes, and there wasn't any kind of short on the 3.9V rail either. The issue occurred identically on both V2 boards' 3.9V regulators (which used the exact same schematic as the previous two boards), but did *not* affect the 3.3V regulator on either board, despite the 3.3V regulator using the same regulator IC and being nearly identical electrically. With this apparent impossibility, we concluded that the issue was most likely to be an issue with PCB fabrication similar to what happened on the RangefinderTest board, where certain traces were connected even though they shouldn't've been. For the time being, because the launch deadline was approaching, we worked around the issue by attaching an external, off-the-shelf regulator to the boards as a 3.9V supply, which got everything booting up. However, we never did isolate the true cause behind this problem.

Thankfully, after this fix, as well as debugging some miscellaneous software problems on the new CPU, the new ground station boards were finally up and running. We completed a full battery of radio tests (Tests 2 through 4), and found that the new boards' performance was far, far better than the transponder had been. The issues with the CMX902 IC were gone, and it was able to reliably output its rated power of +33dBm (and it was also an order of magnitude easier to solder thanks to the new footprint). Receive sensitivity had also improved significantly to about -97dBm – still not quite as good as the SKY65366 on the Ground Station V1, but good enough. All in all, our changes to the radio circuit were a success, and we were finally able to build our own discrete circuit with performance on a similar level to the SKY65366 integrated circuit.

Schematics for the final Ground Station V2 RF amplification circuit follow.


+3V3
C1
10nF 5%
GND
L16
200nH
C2
4.7pF
RX_OUT
RF_LNA_OUT
U7
RF OUT
GND
RF IN
PSA4-5043
CM
CMX902Support.SchDoc
R46
10dBm
TX_IN
TX_ATT
RF_IN
RF_OUT
ENABLE
RF_AMP_OUT
RF_LNA_IN
RX_EN
C50
4.7pF
U4
CTRL
~CTRL/Vdd
RF1
RF2
RFC
PE4259
R14
100k
C36
0.5pF
C37
0.5pF
ANT
This is annoying, but we actually have to REDUCE the output power from the CC1200 with R46 before it goes into the CMX902. Otherwise, the CC1200 can't ever be turned above its lowest power setting, and we lose the ability to lower the transmit power.
The PE4259 RF switch requires all RF lines to have an average/DC voltage of 0V. I believe this is satisfied, so no extra capacitors are needed. C29 in CMX902Support acts as a DC blocker on RF1, and on RF2 C50 acts as a DC blocker.
R14 was also added per Peregrine's AN27 in order to remove any DC charge on the antenna line.
R18
10k
TRX_SW
U8
VCC
TX_EN
SN74AHC1G04DBVR
TRX_SW from CC1200 is connected to GND in TX and allowed to float in RX. TX_EN needs to be high for transmit and low for receive. So, we need some help from a logic inverter.
Title 430MHzAmplifier
Project: GroundStation.PrjPcb
Date: 4/18/2021
File: 430MHzAmplifierV2.SchDoc
Sheet 3 of 15
USCRPL


This circuit taken from section 6 of the CMX902 datasheet
This stuff up here is all for debouncing
Power is supplied to the (beefiest) 3rd stage of the part through L3, so it needs to be a high-current inductor. The current one is just barely beefy enough, but it's the best I can find on Digi-Key. We may have to order one from Coilcraft (who don't sell on Digi-Key) instead.
This is called a Pi-Match circuit. It, and the similar circuit on the other side, is used to match the impedance of the 50 ohm transmission line and the amplifier IC.
ENABLE:
3.3V->Enable amp.
0V->Disable amp.
If between 2.5V and 2.9V, acts as a linear adjustment of output power.
See Figure 4 of CMX902 datasheet.
This section provides the gains for each stage of the amplifier. By varying the left two 3.3V sources from 0 to 3.3V, you can adjust the gain of the amplifier.
+3V9
C38 4.7uF
GND
C39 1uF
C22 470pF
L9 27nH
L11 27nH
C40 1uF
C23 470pF
VDD1
VDD2
+3V3
C27 470pF
C24 470pF
C71 1uF
L12 15nH
RF_IN
C28 470pF
L13 43nH
C41 2.2pF
R15 470 Ohms
C42 1000pF
RFIn
VGS1
VGS2
VGS3
VA
GND
RFOUT
U3
CMX902QT8
CMX_OUT
L14 3.6nH
C29 470pF
RF_OUT
C30 4.7pF
C31 10pF
C33 8pF
ENABLE
R21 68.1 kOhms
R22 68.1 kOhms
C26 10000pF
R20 68.1 kOhms
R23 47 kOhms
C25 10000pF
R19 100 kOhms
R24 39 kOhms
C55 100pF
R6 51 Ohms
Title CMX902Support
Project: GroundStation.PrjPcb
Date: 4/18/2021
File: CMX902Support.SchDoc
Sheet * of *
USCRPL

With all ground stations and the transponder complete, we were finally ready for more in-depth testing. First, we performed several static, long-range tests on the ground (Test 6). Then, a single, long-duration burn-in test was run to ensure that all four ground stations could operate simultaneously for a long time without any conflicts or errors. Finally, after the construction of enclosures for each ground station to keep the elements out (Figure 3.21), we were ready to do a subscale test launch of the system (Test 7)!

Figure 3.21. Four beautiful, completed rangefinder ground stations.

Chapter 4: The Testing Process

A number of tests were run on the Lightspeed Rangefinder system to ensure that it would be able to handle real flight conditions, especially those encountered during a space shot. The most crucial role of these tests was to help us select the best RF settings for use on the CC1200, but they also helped qualify the rangefinder hardware and test the performance of different radio circuits.

This chapter will provide details about each test we ran, including the procedure followed, the test setup used, and the results obtained. Many of these tests apply fairly generally to all digital radio systems, so the procedures we've developed might prove useful for other systems in the future.

Test 1: Jitter Testing

Purpose:

To discover the accuracy of the CC1200-based distance measurement and how this accuracy is impacted by different radio settings.

Test Hardware:

This test was performed on the dual-radio RangefinderTest prototype, since it is the only board with multiple CC1200s attached to the same processor. Though the board was designed for 900MHz transmissions, testing has shown that it can also operate on the 430MHz band with reduced receive sensitivity.

The board was set up with a loopback coax cable connecting the two radios to each other through a -20dBm attenuator.

Procedure:

For each set of radio settings, 300 there-and-back ranging operations were run with a random interval of 0 to 5 us between them (to prevent any synchronization with internal clocks of the CC1200). From each operation, the ranging time was collected and the jitter (the difference between the longest and shortest time), as well as its standard deviation, was logged.

Results:

Settings	**Jitter (ns)**	**Std. Dev. of Ranging Time (ns)**
1. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz	+-3448	3366
2. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz Max-IF	+-3543	3328
3. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz Zero-IF	+-5698	1786
4. 100ksps 2-GFSK DEV=50kHz CHF=208kHz	+-1250	1147

5. 100ksps 2-GFSK DEV=50kHz CHF=208kHz Max-IF	+-1203	1141
6. 100ksps 2-GFSK DEV=50kHz CHF=208kHz Zero-IF	+-1802	1606
7. 500ksps 2-FSK DEV=125kHz CHF=833kHz	+-604	287
8. 500ksps 2-FSK DEV=125kHz CHF=833kHz Max-IF	+-677	245
9. 500ksps 2-FSK DEV=399kHz CHF=1666kHz Zero-IF	+-526	363

Notes:

- "2-GFSK" and "2-FSK" are types of Frequency Shift Keying (FSK) radio modulations. In these modulations, one symbol is equal to one bit of data.
- DEV indicates the frequency DEViation between the center frequency and one of the shifted frequencies.
- CHF indicates the size of the receiving CHannel Filter around the center frequency.

Conclusions:

First, it's important to note that differences of less than about 100ns in the data are not significant; the variation of each configuration between different trials could be in the range of 50 to 100ns.

From this data, it's clear that the primary driver of jitter is symbol rate. This intuitively makes sense: the higher the symbol rate is, the faster the radio module needs to sample the RF data stream in order to demodulate data. At first the relationship seems to be roughly proportional, as tripling the data rate from 38ksps to 100ksps does decrease the jitter by about a factor of three. However, increasing it fivefold to 500ksps only decreased the jitter by about factor of two. Perhaps the CC1200 hits a maximum rate where it can't sample any faster.

The data also shows that the intermediate frequency setting also has an effect on jitter, though more of an opaque one. First of all, changing the IF to the highest non-zero setting (the Max-IF trials) from its default had no statistically significant effect on jitter. However, changing it to zero did have a statistically significant effect, though a different effect for each symbol rate. While it helped produce the best jitter number at 500ksps, it also produced the worst at 38ksps.

The most important question to answer in this test, however, is which RF settings are suitable for ranging. From the data, all of the 500ksps configurations appear workable; all have jitter in the 500-700ns range. Multiplying by the speed of light, this translates to about +-200m single-measurement accuracy: not amazing in isolation, but considering the system can be used to measure distances up to several hundred kilometers, this is not bad! Furthermore, as later tests show, averaging and analysis can reduce this error further.

Test 2: Tx Power Testing

Purpose:

To find and measure the maximum power that each PCB can transmit, which will help determine if the PCBs are functioning as designed.

Test Hardware:

This test was performed on all rangefinder PCBs that were built.

Each PCB was connected through a -40dBm attenuator to a spectrum analyzer. The attenuator served to reduce the PCBs' transmit power to something that the spectrum analyzer could measure.

Procedure:

Initially, power was tested simply by measuring the power of the carrier wave during the rangefinder's normal operation. However, this produced results inconsistent with earlier testing, and further research revealed that this is not the correct method of testing transmit power, since when a radio is modulating, its power is spread out across a wider swathe of the RF spectrum [9].

To correct this error, the CC1200 was specially configured to transmit a non-modulated carrier wave (using its on-off keying modulation mode). To reduce the risk of the amplifier overheating (it is not designed for continuous transmission), the radio was configured to transmit for 1 second and the turn itself off for 5 seconds.

With this mode activated, the amplifier gain of the CC1200 was adjusted until the maximum power was seen out of the receive amplifier.

Results:

Board	**Amplifier**	**Expected Max Output Power**	**Output Power**
RangefinderTest	None	+14.5dBm	+13.8dBm
Ground Station V1	SKY65366	+30.2dBm	+31.0dBm
Transponder (before removal of amplifier)	CMX902	+32dBm	+26.7dBm
Transponder (after removal of amplifier)	None	+14.5dBm	+11.1dBm
Ground Station V2	CMX902	+32dBm	+33.0dBm

From this data, it is clear that the boards we made are reaching roughly their correct transmit powers (except the transponder, due to the amplifier issues discussed in Ch. 3). In fact, most of our boards exceeded their datasheet transmit power – probably because of amplifier manufacturers adding a safety margin on top of the rated transmit power. However, these readings were only achieved after several rounds of fixes. Both issues in both the test setup, such as the spectrum analyzer and attenuator being out of calibration, and issues in the PCBs themselves, such as unsoldered pads on the amplifier ICs, were diagnosed through this testing. In the end, it proved a very valuable way to test the radio hardware before moving on to more advanced testing.

Test 3: Rx Sensitivity Testing

Purpose:

To measure the minimum input power that the radio can receive in each possible configuration, which will confirm that our hardware is working and help determine which configuration to use.

Test Hardware:

Two Ground Station V1 boards were used in this testing because they showed the best sensitivity in testing, and because their output power is controllable between +1.5dBm and +31.0dBm. One ground station unit was placed in a box coated in aluminum foil (to reduce the chance that spurious emissions would connect the two radios) and connected through a fixed, then a variable attenuator to the other ground station unit outside the box (Figure 4.3.1). Signals were then sent between these radios as outlined in the Procedure section.

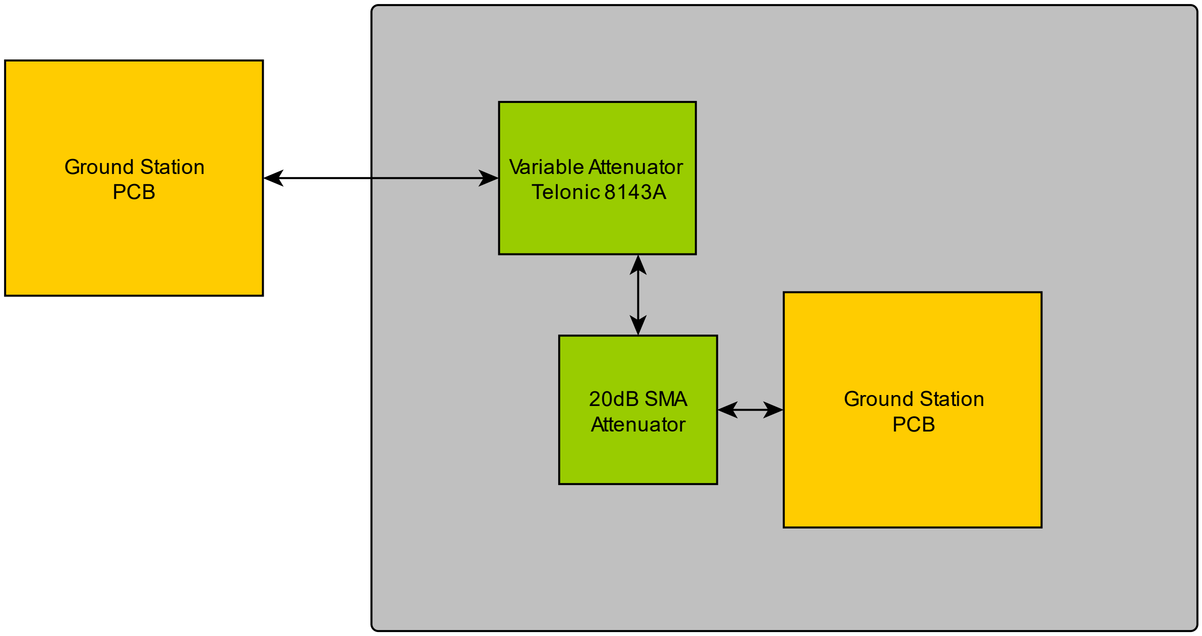


Figure 4.3.1. Sensitivity test setup

Equipment Calibration:

For this test to be successful, precise attenuation of radio signals to very low values was needed. To this end, a used Telonic 8143A variable attenuator was secured (Figure 4.3.2), which is capable of attenuating between 0 and -110dBm and is adjustable in 1dB steps. Since this attenuator didn’t come with a calibration, we calibrated it ourselves using our spectrum analyzer and its tracking generator. Operating at 434MHz, the attenuator and its cables were tested at each 1dB step on the 1dB knob, and at each 10dB step on the 10dB knob down to -60dBm (where our spectrum analyzer hit its noise floor). At all points tested, it was found to be within +-0.5dB of the correct attenuation value once a constant -1dBm offset was applied (it always attenuated 1dBm more than necessary). This known offset was factored out in all subsequent tests.

RF Leakage:

A primary concern in our testing was eliminating RF leakage: specifically, where the attenuator was blocking the signal from traveling between the two ground stations, but the signal found an alternate route and was received anyway. Our testing showed that once the difference between transmit power and receive sensitivity exceeded roughly 115dBm, the signal would “escape” the coax cables and foil box and arrive at the receive radio. Tightening the lid of the box and adding additional layers of foil adjusted this threshold by a small amount, but did not make a massive difference. Unfortunately, due to equipment limitations, more accurate quantification of this effect was impossible. To prevent leakage, the transmit power of the transmitter unit was decreased to its lowest possible value (+1.5dBm), decreasing the power difference in most tests below where the threshold seemed to be. Additionally, for each test, the attenuator was decreased below the sensitivity value until the signal was completely blocked. Since this was possible on all tests after reducing the transmit power (it hadn’t been before), we concluded that RF leakage had been suppressed to a satisfactory amount.

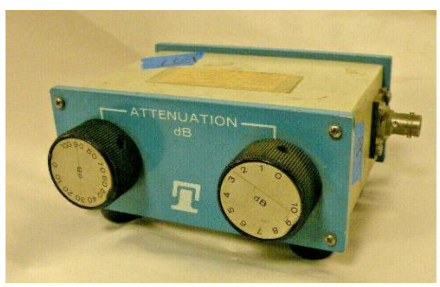


Figure 4.3.2. Variable attenuator.

Procedure:

A program specific to this test was loaded on to each of the ground station boards (See Appendix A). The board inside the box acted as a transmitter and transmitted a sequence of 1408 bytes (11 128-byte blocks) once every second. The stream consisted of alternating 0xAA and 0xBB bytes, so that a byte being dropped or repeated could be detected. The board outside the box received this stream, processed the data, and reported how many bytes were received until there was an error in the data stream. This gave an indication of the byte error rate: how often, on average, the radio would send a byte incorrectly. As we decrease the signal power and it gets closer to the lowest receivable value (the receive sensitivity), the byte error rate will naturally increase as it gets harder for the receiver to distinguish the signal.

When operating the rangefinder sends packets that are six bytes long: four for the sync word, one for the payload, and one for the checksum. So, we set the threshold for sensitivity at a byte error rate of 20: once the average byte error rate across several measurements went below 20, the radios were declared as “not connected”. A byte error rate of 20 would mean that roughly one in three ranging messages would be lost in each direction, so the rangefinder would experience very degraded performance below this point. Yes, this threshold is somewhat arbitrary, but it provides a way to consistently rank the real-world performance of all boards and configurations against each other, which is an important function of this test. It also provides a more conservative performance estimate than if we just used the received power at which no communication is possible, so it’s unlikely to overestimate performance.

The exact location of the byte error rate threshold could be a little bit difficult to identify – sometimes a 1dB step only increased the byte error rate by a little, and the signal varied randomly, so the exact moment when the error rate crossed 20 was somewhat variable. Because of this, readings of sensitivity are only repeatable to about +-1dBm. However, even with this precision limitation, very useful data was obtained from the test.

Results:

Settings	**Measured Receive Sensitivity (dBm)**	**Expected Receive Sensitivity (dBm)***
1. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz	-117.5	-111
2. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz Max-IF	-115.5	N/A
3. 38.4ksps 2-GFSK DEV=20kHz CHF=104kHz Zero-IF	-103.5	N/A
4. 100ksps 2-GFSK DEV=50kHz CHF=208kHz	-109.5	-107

5. 100ksps 2-GFSK DEV=50kHz CHF=208kHz Max-IF	-109.5	N/A
6. 100ksps 2-GFSK DEV=50kHz CHF=208kHz Zero-IF	-100.5	N/A
7. 500ksps 2-FSK DEV=125kHz CHF=833kHz	-100.5	-97
8. 500ksps 2-FSK DEV=125kHz CHF=833kHz Max-IF	-99.5	N/A
9. 500ksps 2-FSK DEV=399kHz CHF=1666kHz Zero-IF	-97.5	N/A

*Expected receive sensitivity was obtained from the CC1200 datasheet for configurations which are listed there [3, p. 11]. Those values are for 900MHz but seem to be similar to 430MHz performance. For other configurations it is marked N/A.

Conclusions:

In general, we were able to achieve receive sensitivities that are roughly in line with the expected values from the datasheet, especially for the 100ksps and 500ksps configurations. This proves that our hardware is working properly and providing optimal performance! Our values were consistently 2-3dBm better than TI specifies. This could be explained by differences in measurement methodology (they do not specify how sensitivity was measured), as well as the presence of the SKY65366 Low Noise Amplifier. In particular, the CC1200 datasheet advertises "Support for Seamless Integration With the CC1190 Device for Increased Range Providing up to 3-dB Improvement in RX Sensitivity" [3, p. 1]. This implies that the presence of an LNA (such as the CC1190 or the SKY65366) should provide up to 3dBm better sensitivity than TI reports in their datasheet, which exactly matches our results.

The sensitivity results at 38.4ksps are something of an aberration, as the measured sensitivity is much, much higher than it should be. We theorize that this is due to RF leakage out of the box (as described earlier) since the power difference in this test exceeds the required threshold for this to occur. If the sensitivity at 38.4ksps follows the pattern of the other tests, the real sensitivity should have been around -114dBm, but we cannot be certain. Thankfully, the 38.4ksps configurations ended up not being considered due to their poor jitter performance, so the 38.4ksps results do not really matter.

These trials provided very useful information about how the CC1200's RF settings affect its performance. As expected, sensitivity is better at lower symbol rates since a worse quality of signal can be decoded by the receiver at lower symbol rates. This difference is fairly significant: we could gain up to 10dBm of additional power margin from switching to a lower symbol rate if we were absolutely desperate for additional link budget. However, as we know from Test 1, these lower symbol rates cause ranging times to be far less accurate, so this was not looked into for now. Also interesting is the effect of changing the intermediate frequency. Zero-IF mode caused significant sensitivity decreases across the board, even when the channel filter was widened as in Configuration 9. Since Zero-IF mode did not show a huge benefit in reducing jitter, it seems like it is not a good option to use going forward. Maximum IF modes, on the other hand, did not show a statistically significant difference in receive sensitivity.

Considering the results of Test 1 and Test 3, we elected to choose Configuration 7 as the radio configuration to use on the rangefinder. This configuration showed quite good ranging jitter and demonstrated the best sensitivity of any of the 500ksps configurations, so it represents a good balance between ranging performance and radio performance. Furthermore, it is one of the officially supported radio configurations offered by TI, so we figured that debugging it and getting support would be easier. With that decided, we moved on to testing the complete system.

Test 4: Full Link Budget Testing

Purpose:

With the transmit power, jitter, and receive sensitivity verified, it was almost time to take the rangefinder into the field. However, one more lab test was still needed to put everything together and determine if the link budget seen in previous tests held up under real usage of the system.

Procedure:

In this test, both rangefinder units were hooked together using the same physical setup as Test 3. However, the unit outside the box was configured as a standard ground station, and the unit inside the box was configured as a transponder. We attempted to increase the attenuation level until the two units could no longer communicate.

Results:

This test did not work out how we intended it to, but still proved to be a very useful qualification tool. When both PCBs were in working order, we were not able to break the link between the two regardless of the attenuator's setting. This is due to RF leakage as discussed previously, and we determined that this occurred if the difference between transmit power and receive sensitivity was over ~115dBm. Since both PCBs were in their final configurations, the power difference should actually have been +138dBm, so this behavior was understandable.

While the test could not tell us what the power difference actually was, it did act as a binary result of "is the power difference over +115dBm", and this turned out to be very useful. Several times during testing, this test did not pass, indicating that the power difference was not actually over +115dBm, even though the transmit power and sensitivity had met expectations when tested individually. Debugging this issue revealed a problem with how the CC1200's settings were being set that only occurred in the full ground station software (some settings had to be altered to transmit Morse code, and were not getting changed back properly). In another instance, the test failed because the malfunctioning CMX902 amplifier on the transponder was discovered to be behaving differently when transmitting short and long messages, which led to our decision to remove it entirely.

Later on, a failed link budget test revealed a serious timing problem. Since the rangefinder relies on sending hundreds of short messages per second, in our initial SKY65366-based design, we made sure to verify that the radio could switch from idle to transmitting in a short amount of time – 369 us for the CC1200 and a further 1.0 us for the SKY65366 [3, 10]. However, when we created the new radio system for the Ground Station V2, timing information for the CMX902 was not available. We had no way of knowing how long it took to begin amplifying once enabled, but this time was miniscule on the SKY65366, so we had to assume it'd be fine. It wasn't fine.

When the CMX902 was acting as a transmitter, it was able to easily exceed +115dBm power difference in the sensitivity test. However, when the same unit was ran through Test 4, it failed, and demonstrated only half that link budget. After experimenting with different settings, we finally found that increasing the "transmit preamble" setting on the CC1200 from 48us to 480us allowed the test to pass. This preamble setting causes the CC1200 to transmit essentially a carrier wave for a certain amount of time before the actual transmission begins, and is designed for exactly this purpose: giving an external amplifier time to warm up before transmitting. Unfortunately, spectrum analyzers can only analyze steady state behavior, so this behavior wasn't visible on there -- whatever the correct equipment is to analyze the behavior of a power amplifier within the first millisecond of it turning on, we don't have it. So, while we could only guess at the exact problem, it seems that the CMX902 requires a fairly substantial amount of time, between 384us and 480us, to fully turn on after being enabled. We don't even know whether it reaches its full output after 480us, or just enough to reach +115dBm power difference. All in all, there are a lot of unknowns about this behavior, and we are hampered from discovering more by lack of test equipment. However, thanks to running this test, we did discover that the preamble setting produces a noticeable improvement in behavior, and used it with the CMX902 hardware in all subsequent tests.

And with that, the rangefinder system had been qualified in the lab to the best of our ability. It was time to take it outside.

Test 5: Walking Tests

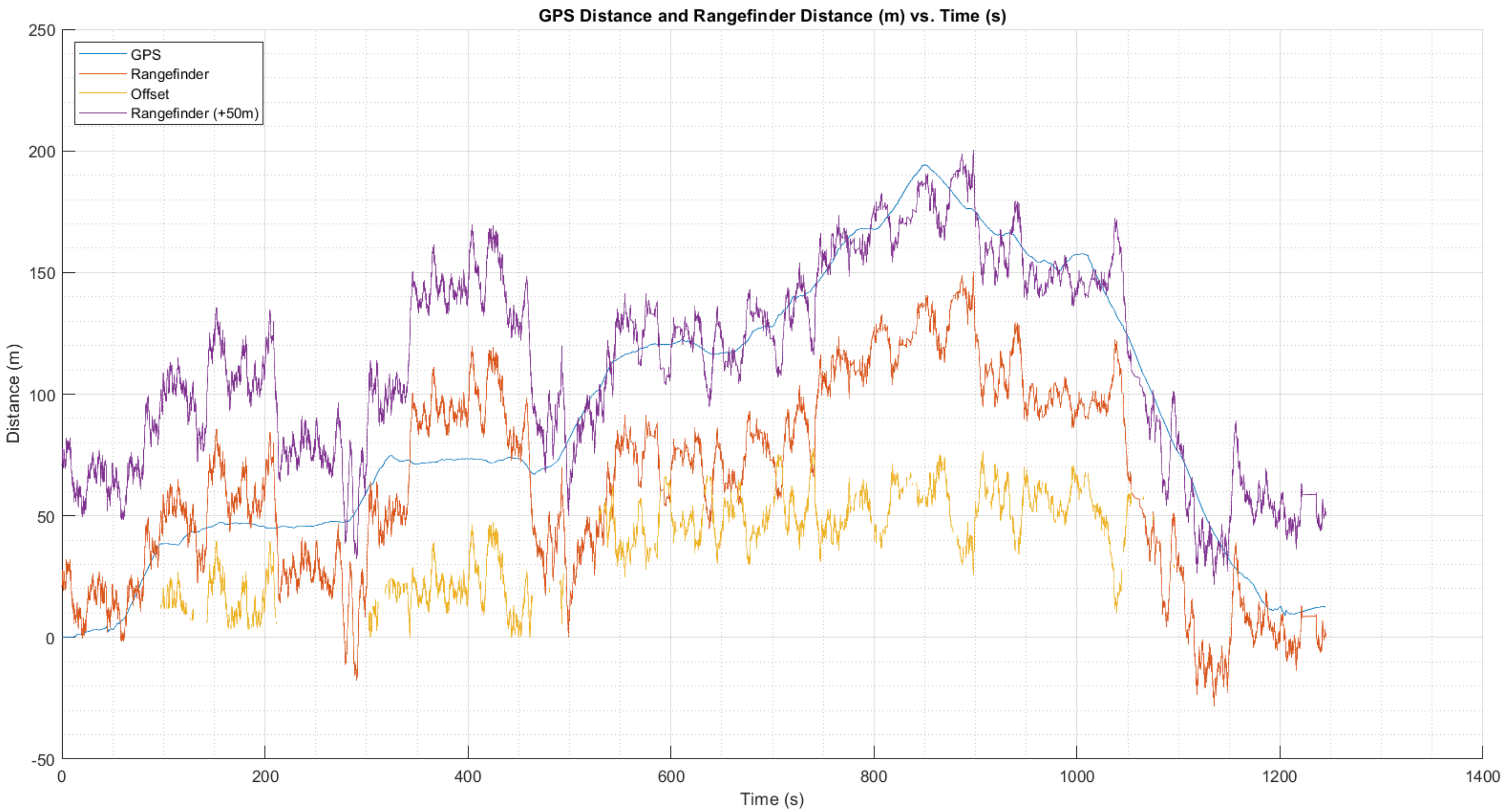
Purpose:

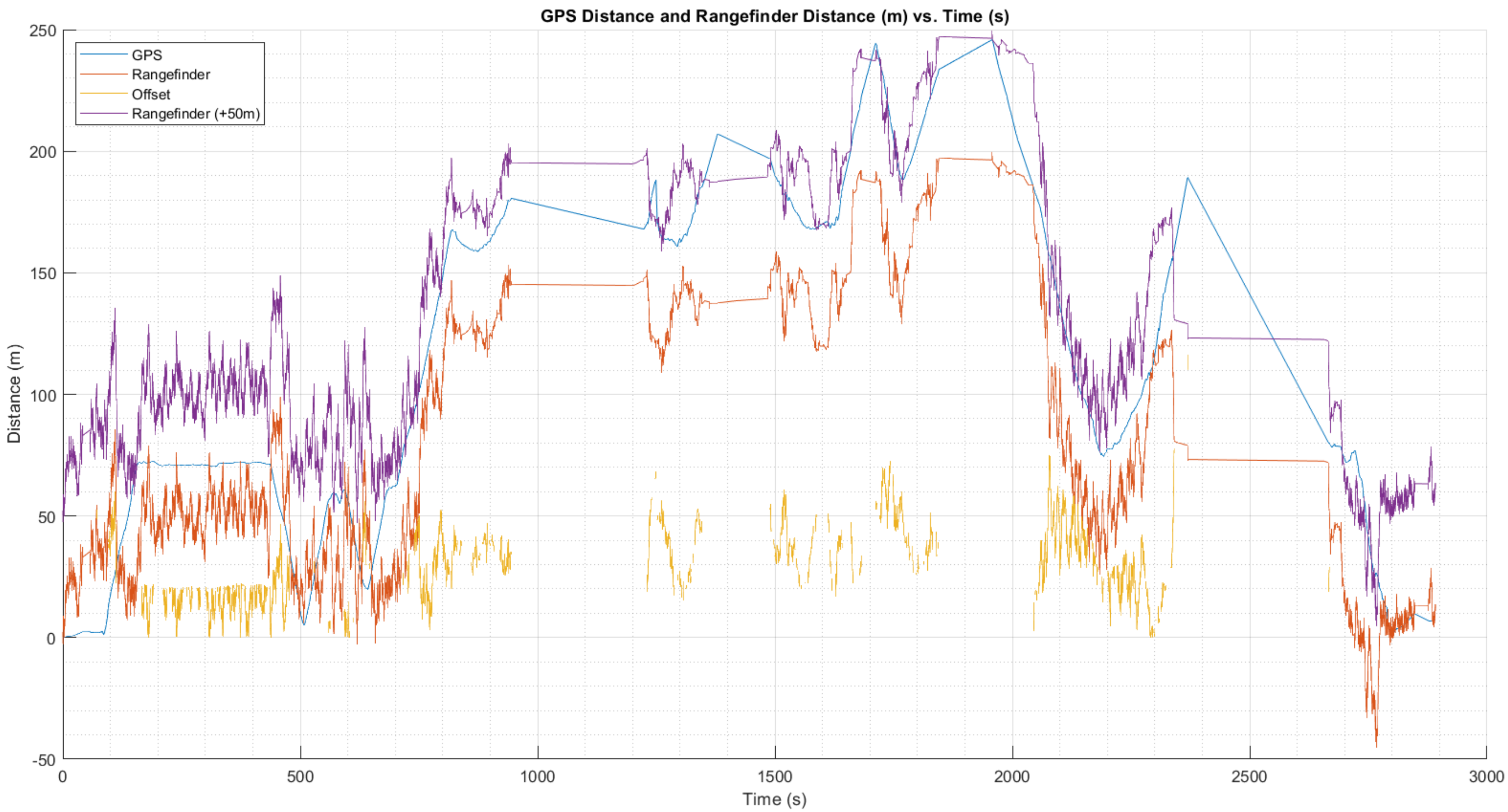
With the rangefinder qualified in the lab, it was time to determine whether or not it would report distance values that made sense physically.

Procedure:

The two first-generation ground stations were taken to the Anstine Nature Preserve in northern San Diego (near where the author was living at the time). One ground station, configured as a transponder, was duct-taped to a wall near the parking lot (to keep it off the ground for better reception), and the other was taken by the author for a walk. Starting from the location of the transponder, it was walked outward until the radio connection dropped, then back nearer the parking lot. This process was repeated twice to create two different datasets. The ground station being walked was recording its position and ranging data to its flash memory, allowing very detailed graphs to be created of the results.

Results:


GPS Distance and Rangefinder Distance (m) vs. Time (s)
GPS
Rangefinder
Offset
Rangefinder (+50m)
Distance (m)
Time (s)
250
200
150
100
50
0
-50
0
200
400
600
800
1000
1200
1400


GPS Distance and Rangefinder Distance (m) vs. Time (s)
GPS
Rangefinder
Offset
Rangefinder (+50m)
Distance (m)
Time (s)
250
200
150
100
50
0
-50
0
500
1000
1500
2000
2500
3000

Analysis:

These graphs show a couple different values:

- GPS: the distance in meters from the starting coordinates, as measured by the ground station's internal GPS.
- Rangefinder: a 101-element moving average of the raw ranging data (which is much less precise), centered on the current time. Note that 101 elements is roughly 10 seconds of data.
- Rangefinder (+50m): Same as Rangefinder, but with all data incremented by a constant 50m.
- Offset: The difference between the GPS and Rangefinder distances

Looking at the data, a few facts were apparent. First of all, there were several times during the second test that the connection to the ground station was lost entirely. These are shown by horizontal lines in the Rangefinder trace, which do not represent real readings. Some range-affecting issues were present in the hardware at the time of this test, and were fixed shortly after. Additionally, we now realize that the location was not ideal for this type of testing, as there were a number of trees, hills, and structures that blocked the signal.

As is visible on the graph, the ranging data did not make sense when the real distance was less than roughly 75 m. This effect surprised us at first, but it turned out to be consistent across all of our trials: the measured distance would increase as physical distance increased up to about 75 m, then would "jump" to a much lower value, then increase again. We believe this is caused by inability of the LNA and receiver on the rangefinder to receive signals *above* a certain power, roughly -30dBm. When the units are relatively close together, the power of the received signal may overwhelm the receiver and prevent it from receiving the primary transmission. So, instead, it locks on to a weaker reflection of the signal that it can receive, leading to an artificially inflated distance value.

Removing all data from less than 75 meters real distance makes the data almost correct: as is visible from the Offset trace, the rangefinder produced a graph with the right shape but consistently 50 meters less than the physical distance. This might look like a serious issue, but it's important to remember the ranging method: besides the time for radio propagation, a constant offset for delay time from the radio modules needs to be subtracted out. In this test, we believe that the offset used was slightly off from the correct value (likely due to the units being calibrated while close together), causing the distance to be skewed downward. As the "Rangefinder (+50m)" trace shows, adding a constant +50m offset causes the rangefinder trace to track the real distance very closely.

Conclusions:

This test helped us learn a little more about our rangefinder system: specifically, that data is not correctly recorded until the rangefinders reach roughly 75 meters away from each other. We also needed to factor this in when calibrating the zero-distance timing offset – we had calibrated the units just by holding them at zero distance from each other. In future testing, we switched to connecting the boards through a high-attenuation attenuator, which prevented the issue.

More importantly, this testing proved, fundamentally, that the rangefinder system works! With no external information apart from the radio signal propagation time, the two rangefinders were able to determine their distance from each other within approximately 20 m of the correct value. This was true across the entire walk, regardless of scenery or distance. This was an extremely encouraging result, and spurred us on to continue development towards the final version of the system.

Test 6: Range Test

Purpose:

Test 5 established that the rangefinder could operate properly over a short distance, no more than 250 m. If the system was going to track the flight of an actual rocket, it would need to work over much longer distances. So, we set out to test whether real-world ranging matched our predictions of accuracy and distance.

Choosing Ground Station Locations:

By March 2021, we knew that we would soon be testing the rangefinder on a launch out of the Mojave Test Area, a high-power rocketry test site in Mojave, CA. So, we set out to choose ground station locations in this area that would be optimal for both pre-launch testing and flight. Careful placement of the ground stations is instrumental to collecting good data from the rangefinder. For example, placing them too close together near the launch site will lead to an almost unimaginably high HDOP (Horizontal Dilution of Precision), and make it impossible to track the rocket's position in the horizontal dimensions (See Chapter 5). We also attempted to find locations that would have the best chance of successful communications with the launch site.

Compared with radio wave propagation through the air, radio wave propagation over the ground is a nasty thing to deal with. There are many different factors that can influence it, such as out-of-phase reflections of radio signals off the ground, reflection and refraction from plants and structures, and even the conductivity of the soil. Compared to free-space path loss alone, radio waves propagating across the ground can lose several tens of dBm more power when crossing the same distance [11, p. 8]. Especially with the reduced link budget of our transponder, this could easily have prevented the rangefinder units from making contact. So, we employed a time-honored way of dealing with this problem: we avoided it.

RPL made use of topological mapping software, CalTopo, to plot the vertical elevation of the ground in the area of the test. We then identified a number of points from which radio signals could propagate to the launch site without passing directly over the ground for most of the distance (Figure 4.6.2).   Finally, we filtered these points based on what areas could be easily driven to, and settled on two ("Charlie" and "Delta") that were far away enough to produce good data, but were easily accessible and likely to have good radio propagation to the launch site (Figure 4.6.1).

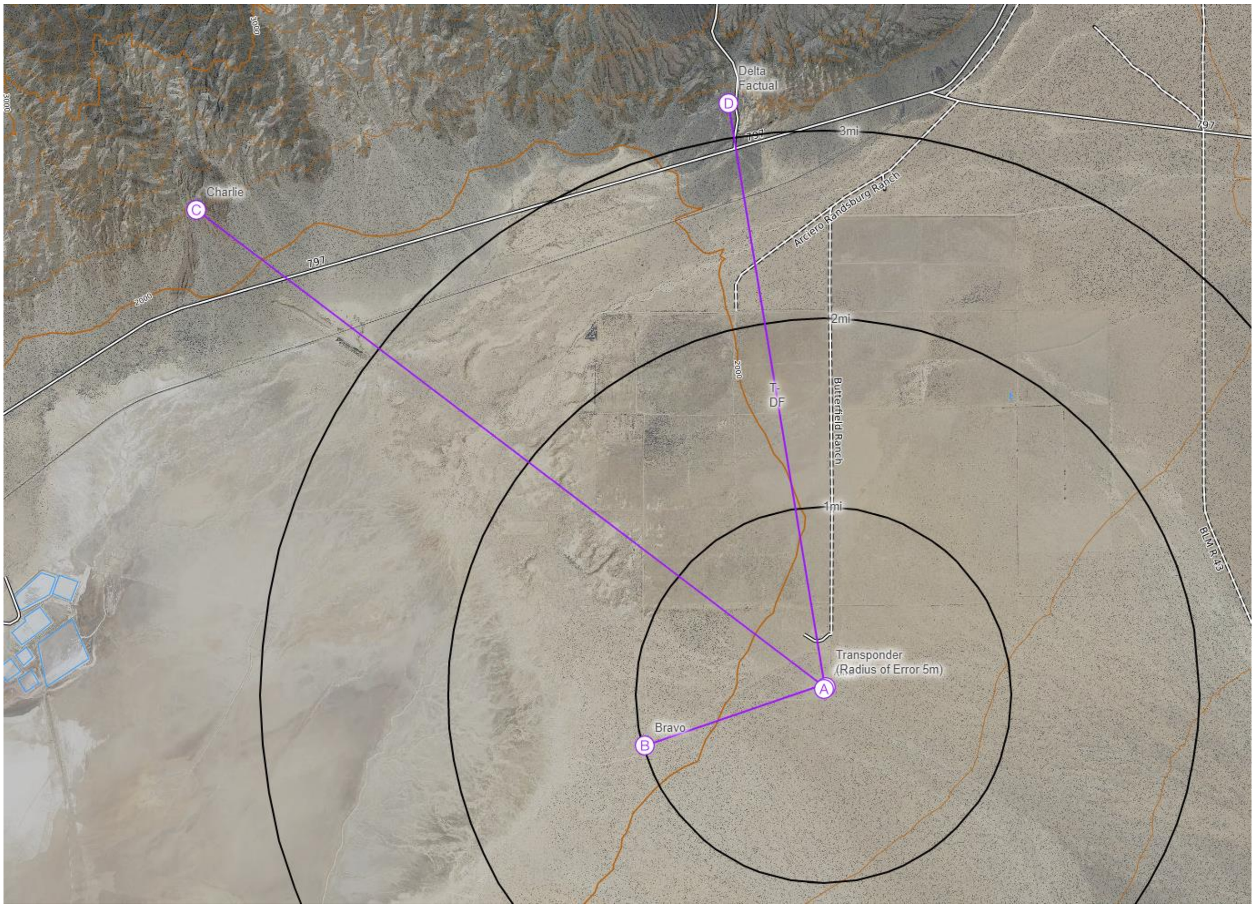


Figure 4.6.1. Ground station locations for Mojave Test Area launches. https://caltopo.com/

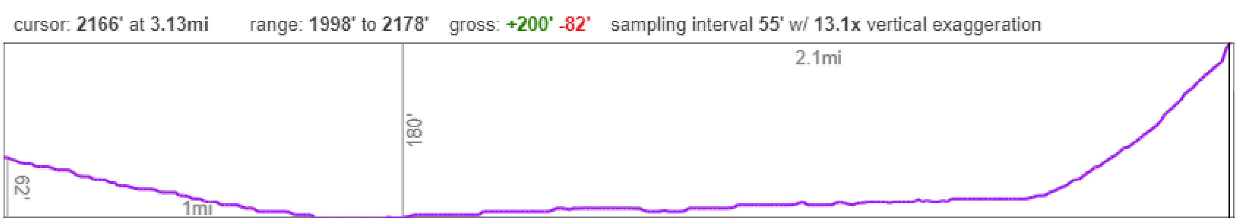


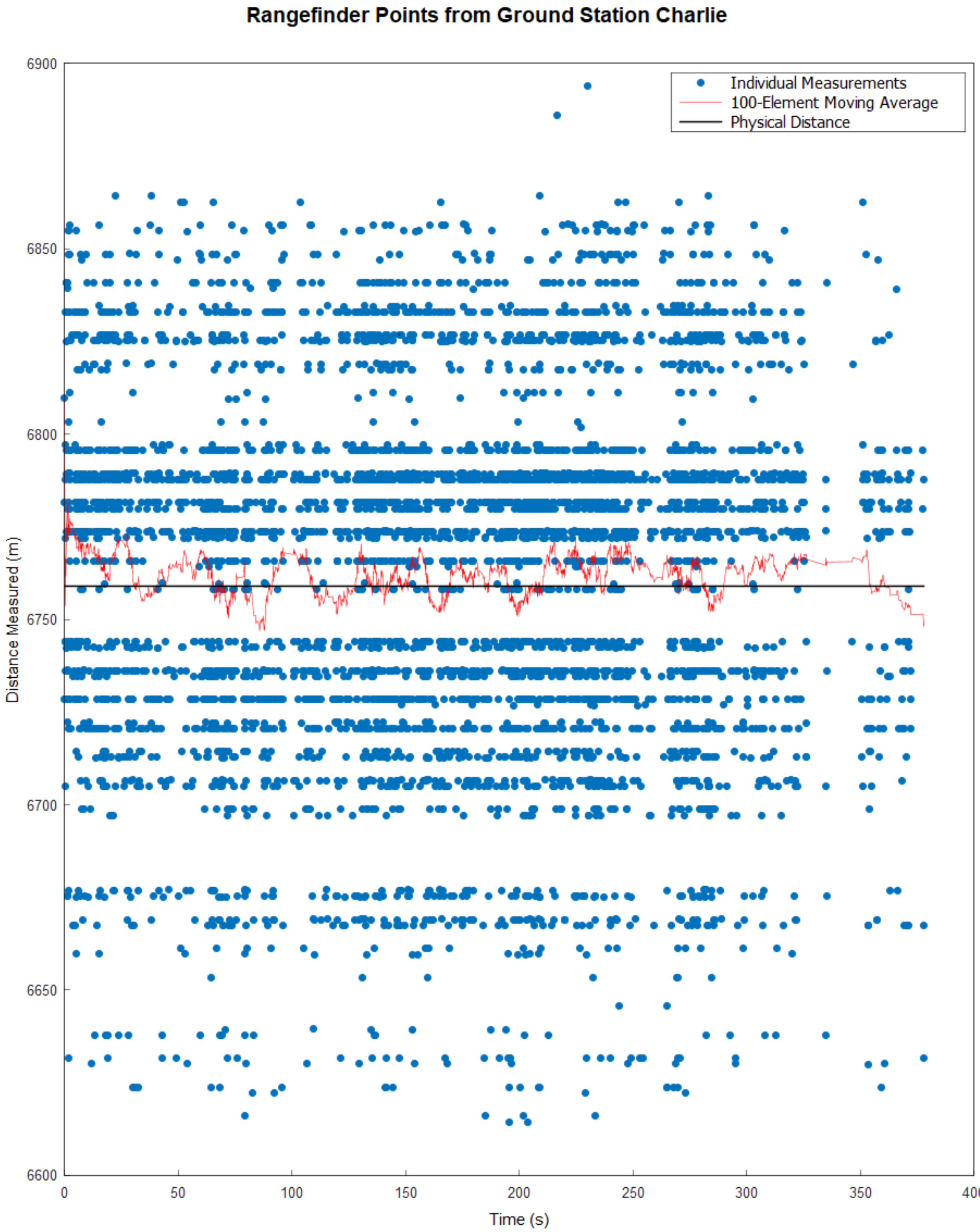
Figure 4.6.2. Plot of ground height between the launch site (left) and Ground Station Delta (right). https://caltopo.com/

Procedure:

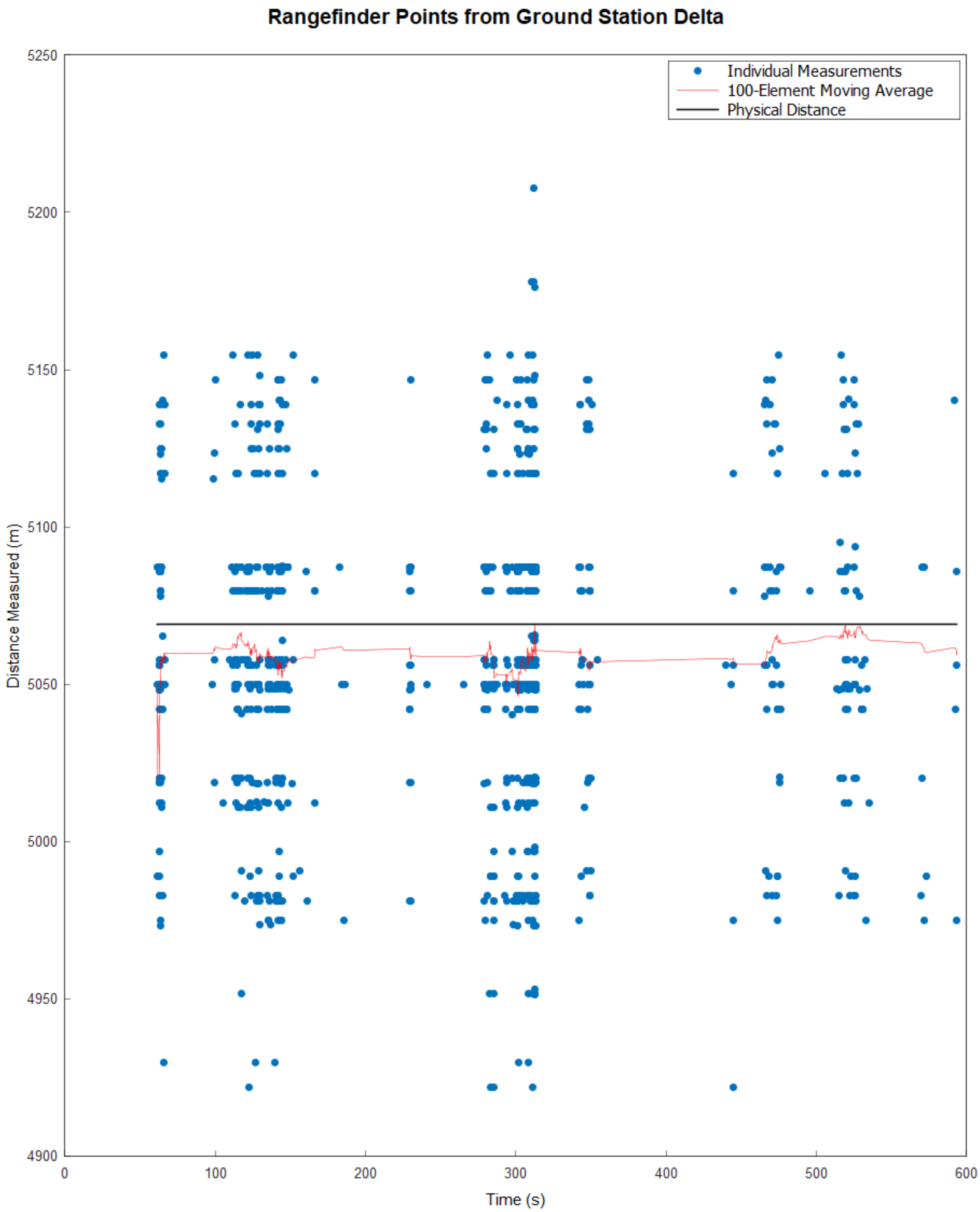
One warm Saturday afternoon, several RPL members headed out to the desert to conduct the preliminary range test of the rangefinder. Two teams of ground station operators (GSOs) headed out to points Charlie and Delta above. Each team carried a Ground Station V1 (because of its better receive sensitivity), and one was equipped with a 13dBi Yagi antenna while the other was equipped with just a 2dBi monopole, so that we could test their relative effectiveness. Meanwhile, a third team stayed at the launch site and operated the transponder, which had already been integrated into a section of the body tube for the launch vehicle.

Once everything was set up, it was time for the moment of truth: activation of the system. Once the transponder was turned on and positioned, each ground station was enabled and allowed to collect data for a few minutes. The results are as follows:

Results:



Rangefinder Points from Ground Station Charlie
Individual Measurements
100-Element Moving Average
Physical Distance
Distance Measured (m)
Time (s)
6900
6850
6800
6750
6700
6650
6600
0
50
100
150
200
250
300
350
400


Rangefinder Points from Ground Station Delta
Individual Measurements
100-Element Moving Average
Physical Distance
Distance Measured (m)
Time (s)
5250
5200
5150
5100
5050
5000
4950
4900
0
100
200
300
400
500
600

Analysis:

This test went extremely well, and confirmed many of our predictions about the rangefinder system. Distances recorded by the system were accurate, especially on Ground Station Charlie, where the average was correct to within a few meters (at a distance of 6.8 km) and the error at any point didn't exceed 20 meters. Delta's data followed the same distribution but was skewed roughly 10 meters downward from the correct distance. This could be due to error in measuring the real distance (regrettably, the position of the transponder was not recorded very accurately), or some type of systemic error inherent to the CC1200 radios under those specific conditions. However, this error is small compared to the rangefinder's +-180m single-measurement error, so we do not feel that it is very significant.

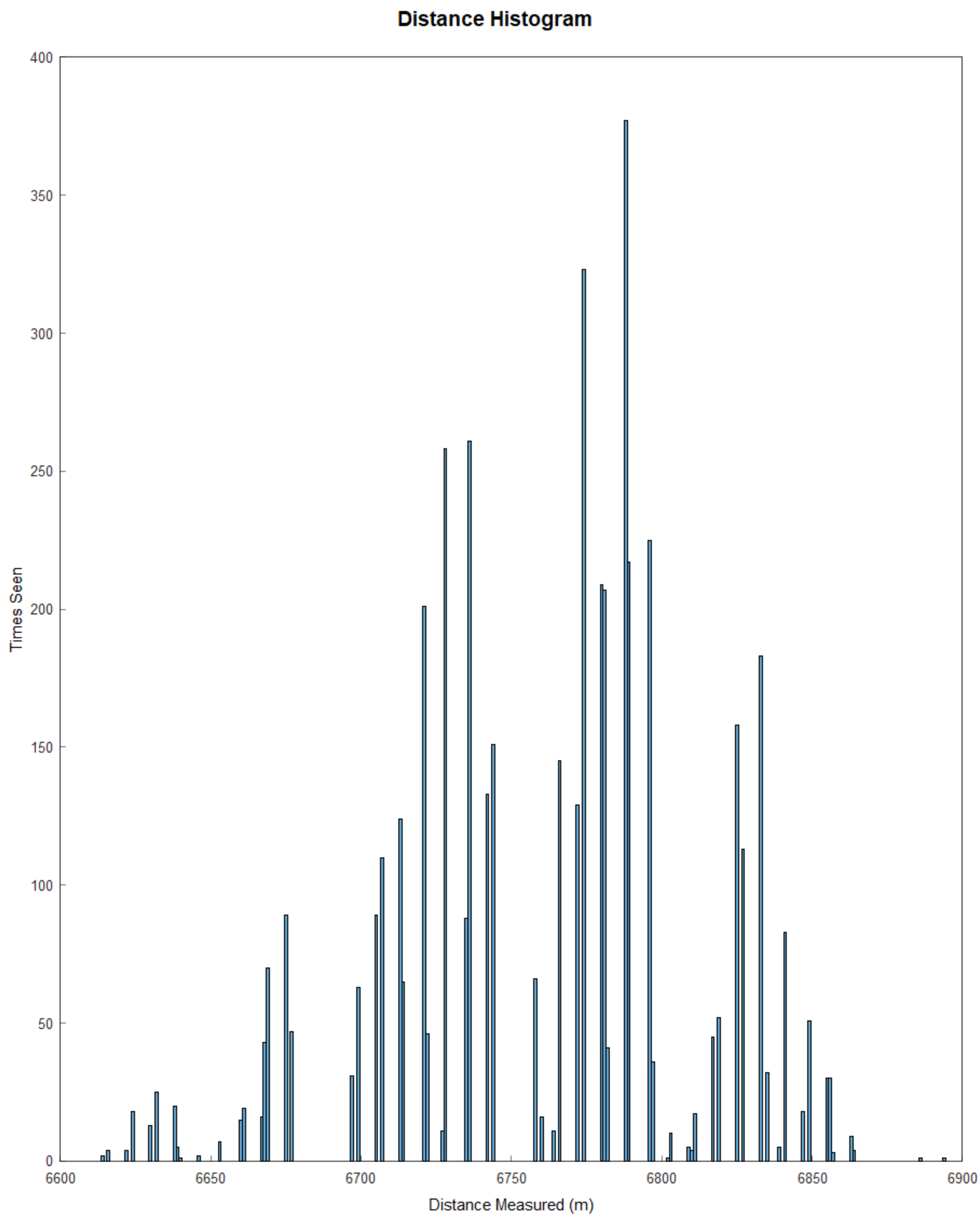


Figure 4.6.3. Histogram of raw measurements from Ground Station Charlie.

Also interesting is the distribution of raw points around the average. These effects are most easily seen in a histogram (Figure 4.6.3). While Test 1 had calculated the jitter of the system as

+-180m, in this test it was far more precise: the vast majority of points were within +-100m of the average, and the standard deviation was only 49.6m. This could be attributed to, as described in Test 5, radio wave reflections that were present in the lab but absent in long-range use. Additionally, points were neatly quantized into distinct distances roughly 9 meters apart. This indicates that the timing data is being “ticked” by a clock inside the CC1200 once every 60 nanoseconds, or at 16.6MHz.

By comparing to the theoretical link budget of the system (readjusted for the as-built transponder and the real Yagi antenna) (Figures 4.6.4 through 4.6.7), we can also establish some facts about the RF performance of the transponder. The system cannot be fully characterized, but some lower bounds can be established.

Stage	**RF Power**	**Notes**
Transmit Power	+11.1dBm	Output of jumper-wired CC1200 on the transponder
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors
Transponder Antenna Gain	0dBi	Assuming omnidirectional antenna on transponder
Ground Antenna Gain	+13dBi	Using Diamond A430S10 yagi antenna
Receiver Loss	-1dBm	Ballpark loss from cables and connectors
Receiver Sensitivity	- (-100.5dBm)	Sensitivity of Ground Station V1
Path Loss	-101.9dBm	*f = 442MHz, d = 6.75 km*
Power Margin	**+20.7dBm**	

Figure 4.6.4. Link budget for the test when the transponder was transmitting to Ground Station Charlie.

Stage	**RF Power**	**Notes**
Transmit Power	+31dBm	Output of SKY65366
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors
Ground Antenna Gain	+13dBi	Using Diamond A430S10 yagi antenna
Transponder Antenna Gain	0dBi	Assuming omnidirectional antenna on transponder
Receiver Loss	-1dBm	Ballpark loss from cables and connectors
Receiver Sensitivity	- (-81.5dBm)	Sensitivity of transponder
Path Loss	-101.7dBm	*f = 442MHz, d = 6.75 km*
Power Margin	**+21.8dBm**	

Figure 4.6.5. Link budget for the test when Ground Station Charlie was transmitting to the transponder.

Stage	**RF Power**	**Notes**
Transmit Power	+11.1dBm	Output of jumper-wired CC1200 on the transponder
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors
Transponder Antenna Gain	0dBi	Assuming omnidirectional antenna on transponder
Ground Antenna Gain	+2dBi	Using Diamond SRH77CA monopole antenna
Receiver Loss	-1dBm	Ballpark loss from cables and connectors
Receiver Sensitivity	- (-100.5dBm)	Sensitivity of Ground Station V1
Path Loss	-99.5dBm	*f = 442MHz, d = 5.07 km*
Power Margin	**+12.1dBm**	

Figure 4.6.6. Link budget for the test when the transponder was transmitting to Ground Station Delta.

Stage	**RF Power**	**Notes**
Transmit Power	+31dBm	Output of SKY65366
Transmitter Loss	-1dBm	Ballpark loss from cables and connectors
Ground Antenna Gain	+2dBi	Using Diamond SRH77CA monopole antenna
Transponder Antenna Gain	0dBi	Assuming omnidirectional antenna on transponder
Receiver Loss	-1dBm	Ballpark loss from cables and connectors
Receiver Sensitivity	- (-81.5dBm)	Sensitivity of transponder
Path Loss	-99.5dBm	*f = 442MHz, d = 5.07 km*
Power Margin	**+13.0dBm**	

Figure 4.6.7. Link budget for the test when Ground Station Delta was transmitting to the transponder.

Since the system was able to connect to Ground Station Charlie, we can calculate from the Friis Transmission Equation that the radio performance between the two units was not more than 20dBm worse than predicted – it was at least 102dBm before path loss is subtracted. The same analysis can be done with Ground Station Delta: the unit was able to operate correctly with only 12.1dBm of power margin, indicating that our power predictions were less than 12.1dBm off. Were we to do the test again, it would be smart to bring a variable attenuator with us to the remote sites and manually attenuate the signal until it dropped off completely. This would be an effective further verification of power margin.

Also interesting was the fact that poor antenna orientation caused Ground Station Delta to lose lock for some periods of time, as the operators tested out a number of different antenna orientations. Though this is a very rough estimate, if we assume that poor antenna orientation can incur at most a 20dB penalty to link budget, the power margin of the system could not have been more than 20dBm. This is in line with the calculation in Figure 4.6.7, and helps confirm our link budgeting process.

All in all, this test confirmed many of our hypotheses about the rangefinder's accuracy and range, and made us finally feel ready for a real launch.

Test 7: Lumineer Launch

Purpose:

After all of the previous qualification tests had concluded, we finally felt that the rangefinder was ready to take to the skies. All we needed was a vehicle, but due to the ongoing pandemic we could not be certain about launching one of our own anytime soon. So, Rocket Propulsion Lab forged an agreement with model rocketry company bps.space for the rangefinder to hitch a ride on their experimental Lumineer launch vehicle, which took to the skies on April 10, 2021. Unfortunately, we did not get the flight data we had been hoping for, but we did learn a lot about how to integrate the system with a rocket launch, and obtained some interesting new failure modes to look into.

Procedure:

In preparation for the launch, RPL developed a detailed launch-day procedure for the rangefinder system, which specified where the ground station operators needed to go, determined when they needed to be there, and established a checkin system for communications between them and the launch site. Integrating these procedures with bps.space's own launch script took some thinking, but eventually we agreed on a procedure that worked for everyone.

Since the range test had gone well, we made use of the same "far" ground station positions, with two additional "near" ground stations: Alpha directly at the launch site, and Bravo one mile to the west. This formed a quadrilateral shape which we believed would provide accurate positioning information up to a reasonable altitude. Since signals from these near ground stations did not have to travel as far, they were given the Ground Station V2 units with their slightly worse sensitivity.

On the morning of the launch, procedures went smoothly. All of the ground station operators drove to their assigned locations, set up the equipment, and switched it on. As expected, their call signs were heard on the ham radio unit back at base, indicating that their ground stations were in working order. However, none of the ground stations were able to receive any pings from the transponder in the vehicle except for Ground Station Alpha, which was less than a hundred meters away from the launch pad. While this was ominous, there was a plausible explanation: the vehicle was mounted on a large metal launch rail which was, unfortunately, positioned between the transponder and the far ground stations. We believed that this could plausibly cause enough signal loss to prevent the rangefinder from working. Weeks prior, we had considered the case where ground stations could not connect before launch, and decided that it was likely to be a geographic issue, and would hopefully be resolved once the vehicle was in the air. So, we followed our predetermined plan, which was to continue with the launch under the assumption that the issue was temporary. In retrospect, we probably should have reevaluated this decision in light of the success of Test 6…

Results:

Lumineer launched, taking the rangefinder unit to an altitude of over 10km. However, the rangefinder was only able to maintain connection to Ground Station Alpha for a small portion of this flight – not more than 3 seconds after ignition (Figures 4.7.1, 4.7.2). Despite quite a bit of scanning with the antenna, connection was never restored, even during the vehicle's descent phase. Furthermore, ground stations for Lumineer's other radio systems also lost communications during the ascent and were only able to receive a few sporadic packets for the remainder of the flight.

While we on the ground were puzzling over connection loss, another problem occurred in the air. During descent, Lumineer's drogue parachute failed to deploy properly. This caused it to impact the ground very hard, damaging the avionics bay and making a post-mortem analysis of the issue, at least the electronics aspects of it, significantly more difficult. While we had intended to replace it anyway, the transponder PCB was damaged badly enough from impact that no more testing operations were possible for the semester (Figure 4.7.3).

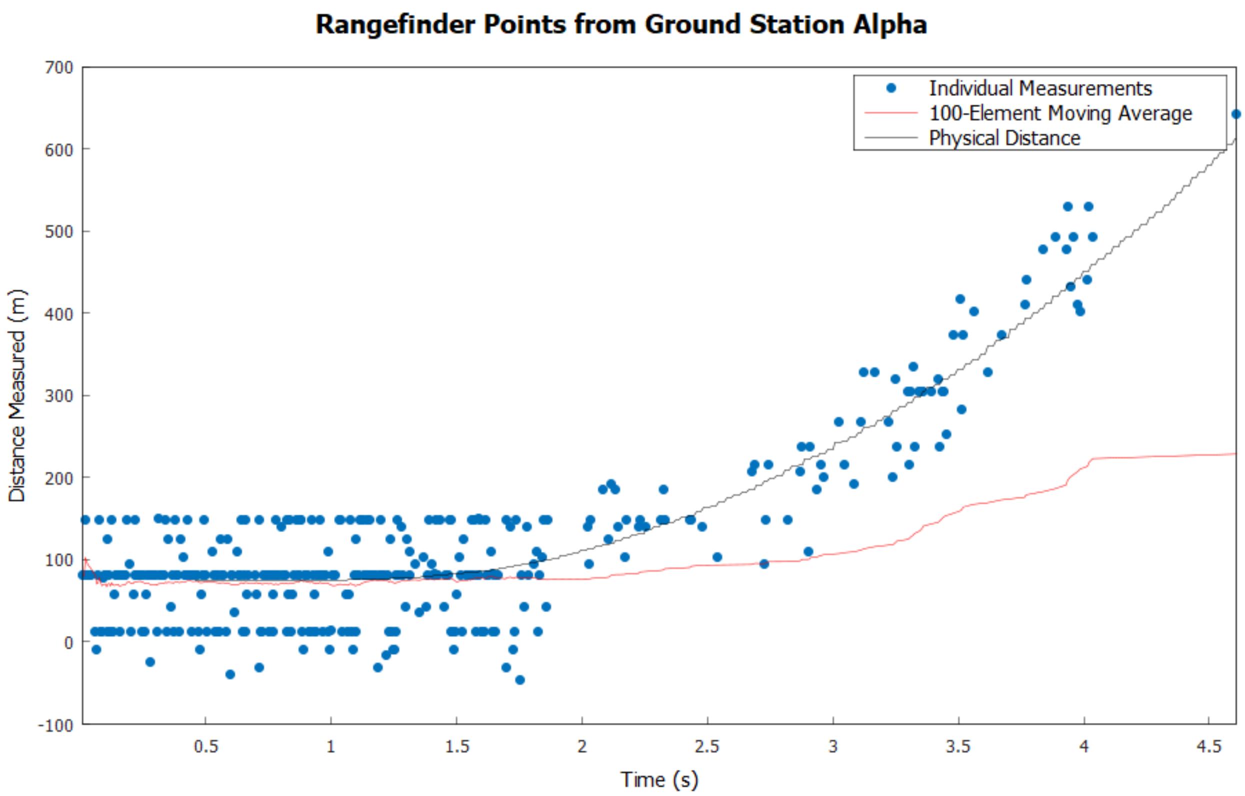


Figure 4.7.1. Raw points recorded during the Lumineer launch from Ground Station Alpha. Note that the moving average (red line) is expected to lag behind the data points by a few seconds.

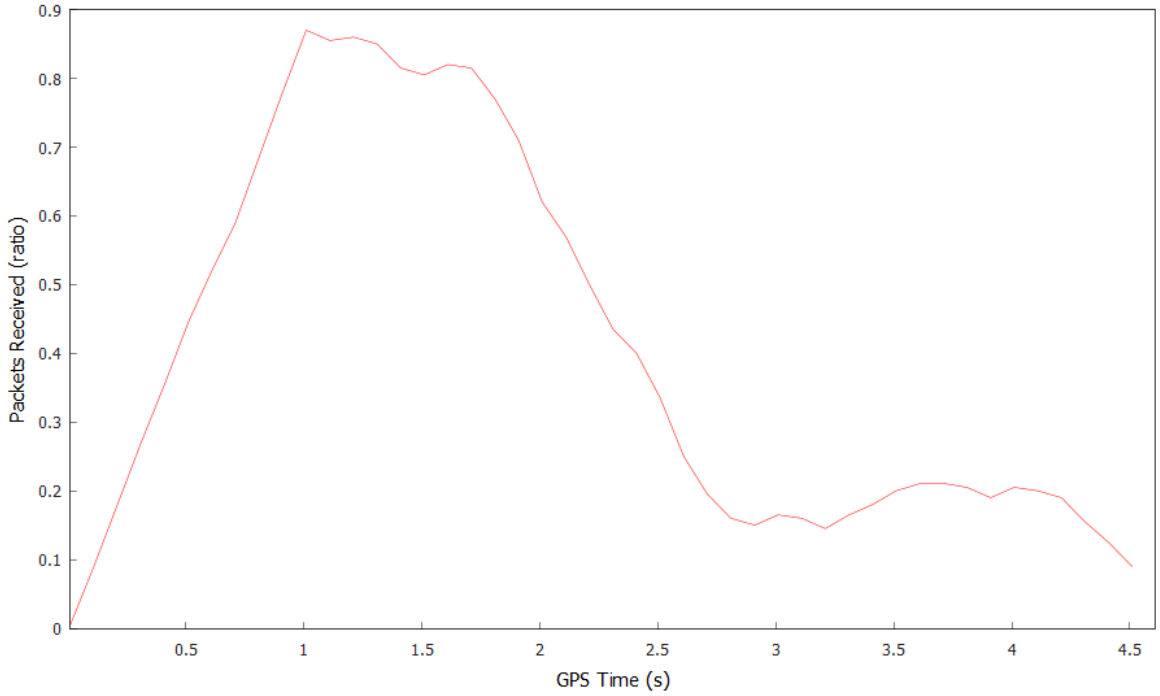


Figure 4.7.2. Percentage of sent packets received by Ground Station Alpha (nominal transmission rate 200 Hz). One second moving average.

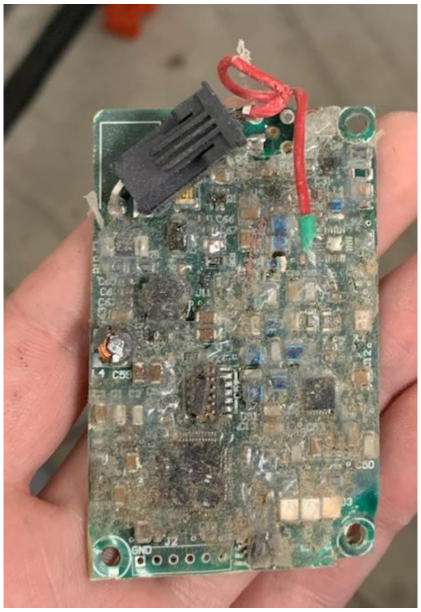


Figure 4.7.3. Transponder PCB after recovery from the crashed vehicle. Note that prior to flight the PCB was coated in silicone potting compound to protect it from G-forces and condensation.

Analysis:

Data collected by Ground Station Alpha appears to fit the expected flight path of the vehicle, and the measurement error is in line with that seen during Test 6, so it appears that while connection was maintained, the system worked correctly and as it had in previous tests. The more interesting question to answer is: why did the rangefinder lose connection when it had worked correctly under nearly identical conditions in Test 6?

The plot thickens further in light of issues seen on the other radios seen on the Lumineer vehicle. The vehicle also contained a Telemetrum standalone unit and an XBee radio for telemetry from the main avionics unit, and both of these radios lost consistent communications before the vehicle had ascended more than a few kilometers. Both of these radios' power margins were even higher than the rangefinder, so they should have easily been able to connect for the whole flight. These multiple failures suggest two different possible explanations for the flight: that there was an electrical failure in the avionics bay, or that there was some kind of radio attenuation or interference created by other elements of the rocket.

The Lumineer vehicle used a separated avionics bay design, with multiple radios in a lower bay connected via cables to the batteries and main avionics unit further up (Figure 4.7.4). No audit was made by RPL of the connectors and cabling leading to this area, so it's possible that wiring and connectors were used that could have failed under flight loads. Though the connectors (a screw-lock type) were verified still plugged in after the vehicle landed, high vibration and acceleration forces could have momentarily, or even permanently, disconnected them electrically during flight. This hypothesis is supported by the fact that all radios dropped out at some point during the ascent stage, where g-forces are some of the most violent. RPL has had a number of electronic failures from g-forces in flight, and can attest that these issues are hard to predict or avoid. However, a challenge to this theory is that not all radios lost communications at the same time; their communications seemed to drop in order from weakest to strongest. That would suggest a problem with radio propagation itself.

Figure 4.7.4. Rangefinder unit (highlighted) next to the XBee radio in the CAD for Lumineer.

To explore this theory, a link budget was created for each of the radio systems on Lumineer, as well as the two "near" ground stations. This was used to obtain the theoretical power margin at the design apogee of 10km. Using the Friis Transmission Equation, the actual power margins were also calculated based on the distance when communication was lost (by assuming that communication was lost at the point where power margin became zero).

Radio System	Theoretical power margin @ 10km altitude	Altitude when last packet was received	Power margin this indicates @ 10km altitude	Apparent power loss
XBee	30.3dBm	3.7km	-9dBm	39dBm
Telemetrum	29.3dBm (approx.)	1.5km (approx.)	-17dBm	46dBm
Rangefinder GS Alpha	3.6dBm	650 m	-23dBm	27dBm
Rangefinder GS Bravo	14.6dBm	(no packets received)	< -16dBm (if it had been > -16dBm, GS Bravo could've connected)	>=30dBm

If we take it as a given that the loss of communications during ascent was from RF power loss, and our previous power margin tests were accurate, there was clearly a monumental amount of loss occurring during this launch – all radios lost at least 27dBm of power compared to their theoretical power margins. The hardest hit was the Telemetrum system, which lost communications at an extremely low altitude relative to its power margin. Comparatively, the rangefinder did better than the other systems, however all the radios performed anomalously. There are a couple of possible reasons for this, though no smoking gun has yet been identified. All ground antennas were nearly directly below the vehicle during its ascent, which is a very poor orientation for a monopole antenna to transmit in. This could have potentially explained 10-20dBm of loss during ascent, but this issue should have disappeared during the vehicle's descent phase where it was in a tumbling fall. Additionally, the other ground stations had a much better side view of the vehicle, and as such should have seen far less loss from antenna orientation. However, none of the remote ground stations were able to connect either. This suggests that antenna loss could not have been the only factor at play.

Structures on the rocket itself could also have contributed to the issue. The fuselage of the rocket itself was made from fiberglass, which should not have affected radio signals significantly – RPL's own nose cones are made from similar materials for RF permeability. However, the paint and primer that the fuselage was painted with is more of an unknown, and could potentially have attenuated radio signals. More research into this is in progress. Structures inside the rocket could also have caused problems: A large, powered-down reaction wheel motor (containing sizeable permanent magnets) was located directly above the transponder and XBee, and the rocket motor (which used a large metal cylinder as its motorcase) was placed a few inches below the tip of the antenna. Though it is difficult to say definitively, these structures could have further blocked and attenuated radio signals, especially in the downwards direction. Going forward, we may attempt to create a mockup of the as-flown lower avionics bay using the same materials and test its effect on the amplitudes of radio signals in different directions.

Figure 4.7.5. Exploded diagram of Lumineer's lower avionics bay.

Lastly, we considered the possibility of interference between radios on Lumineer itself. The Telemetrum and the Rangefinder are only 7 MHz apart, so it's possible that the Telemetrum transmissions blocked the transponder from communicating, or at least weakened it somewhat. However, after signal was lost the transponder would have stopped transmitting, so the other radios should not have been affected. Furthermore, the XBee unit was operating at nearly twice the frequency of the rangefinder and Telemetrum, so it could not have been affected. Our conclusion is that interference could potentially explain issues with the transponder, but not issues with the other radios on the vehicle.

At present, we still do not have a definite explanation for what happened to the radios, including the Lightspeed Rangefinder, on the Lumineer launch. There are a number of explanations for reduced signal power, especially during ascent, but these explanations are not proven yet, and they struggle to explain even the XBee system, with its monumental power margin, losing communications. Meanwhile, an electrical failure accounts for a total loss of signal, but does not explain the order that the radios lost lock during ascent. One possible theory is that both issues occurred at the same time: RF losses were encountered during ascent, and then an electrical failure occurred and prevented the radios from regaining contact during apogee and descent. Historically, on Rocket Propulsion Lab's vehicles, electrical failures have been most likely to occur during parachute deployment, which is when the vehicle sees the most mechanical stress and force. A connector coming loose or a wire failing at the same point on Lumineer's launch would not be unlikely, and might produce something similar to the effects seen on the launch.

At the end of the day, while this launch did not turn out how we expected, we still obtained useful information from it. First and foremost, we developed launch procedures for using the rangefinder to track a rocket launch, and successfully executed them with almost no delays. Additionally, we successfully integrated the rangefinder system into a rocket that was not our own, got it to the pad, and launched it. While the radio system didn't function as intended, this appears to be an issue at the vehicle level, affecting all the radios, rather than an issue with our system itself. We only earned a small sliver of data, just enough to show the beginning of the vehicle's ascent, but that small sliver means a great deal. With time, we will try again, and we are confident that we can develop that small sliver into a detailed picture of a rocket's whole flight.

Chapter 5: Data Analysis

As we built the rangefinder, we needed to make sure that we had a way to analyze the data that we would be gathering. So, experiments were run to find the best method of processing the bevy of points gathered by each ground station and converting these into a coherent, three-dimensional trajectory. Compared to other aspects of the system, this process is still in its earlier stages, and we have not yet reached a final method to use for space shots. However, we have prototyped several techniques that show promising results.

Predicting Error: Dilution of Precision

One of the first problems arising in the usage of the rangefinder is where to place the ground stations in order to get optimum performance out of the system. A number of different factors need to be considered, such as line-of-site to the launch location (so you can tell if it's working before launch), ground distance from the launch site (which determines how much driving you need to do on launch day, and how much extra RF performance you need), and the effect of the chosen locations on accuracy of the system. However, balancing these factors is more difficult. How much separation is enough to ensure accurate measurements? At what distance are there diminishing returns? To answer these questions, a more scientific analysis is needed.

Thankfully, the process of calculating the error of a trilateration system is well studied thanks to research into GPS positioning. The key is a calculation called *Dilution Of Precision*, or DOP, which gives the ratio between the standard deviation of each range measurement and the standard deviation of the resultant position. DOP can be calculated for each cardinal axis (north, east, and upward), as well as Horizontal DOP (HDOP), which combines DOP in the north and east axes, and Position DOP (PDOP), which provides DOP in three dimensions [12]. Simulations of DOP for a flight can help guide the placement of ground stations based on acceptable levels of error. For instance, if ground stations are clumped very close together on the ground below the vehicle, this actually decreases VDOP slightly, essentially because the measurements of height are being averaged together. However, this arrangement causes HDOP to skyrocket, because small variations in the measured range could cause huge shifts in its estimated horizontal position. In tested configurations, horizontal error can be in the range of several kilometers at apogee. Fortunately, for amateur rocketry applications, tracking a rocket's vertical position is far more important for measuring performance (and snagging records) than its horizontal position, so this tradeoff is acceptable.

Figure 5.1 demonstrates the concept behind HDOP and VDOP (Vertical DOP), simplified to 2D. Consider that if r_2 is held constant, r_1 determines the calculated position of the rocket. With the rocket far above the ground stations, a small error in r_1 would produce very significant changes in the rocket's horizontal position, but only small changes in its vertical position. For this reason, the HDOP in this configuration is much higher than the VDOP. If the ground stations were to be spread out wider on the ground, this would improve HDOP, but at the cost of increased VDOP.

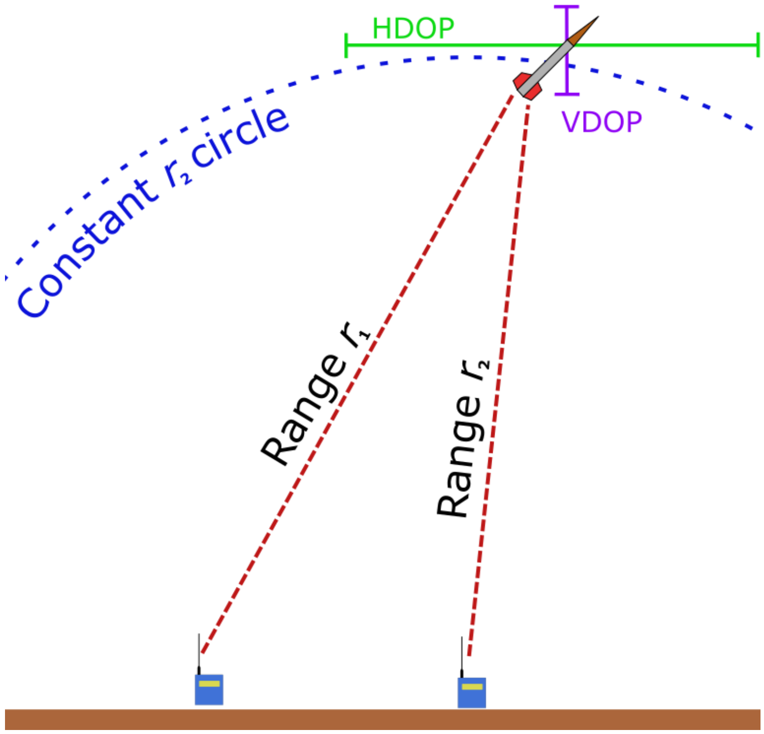


Figure 5.1. Diagram of ranging and dilution of precision.

To test the rangefinder's viability on high-altitude flights, we plotted the DOP values for a simulated flight of RPL's upcoming space shot, Domepiercer, which was projected to reach an apogee of about 440,000 ft (134 km). In this simulation (Figure 5.2), we assumed that there were 4 ground stations in a square arrangement 2 mi (3.2km) on a side, centered on the launch site. This was simply an initial guess and will certainly be refined in the future, especially since Test 6 showed that ground station placements of almost four times this distance are feasible.

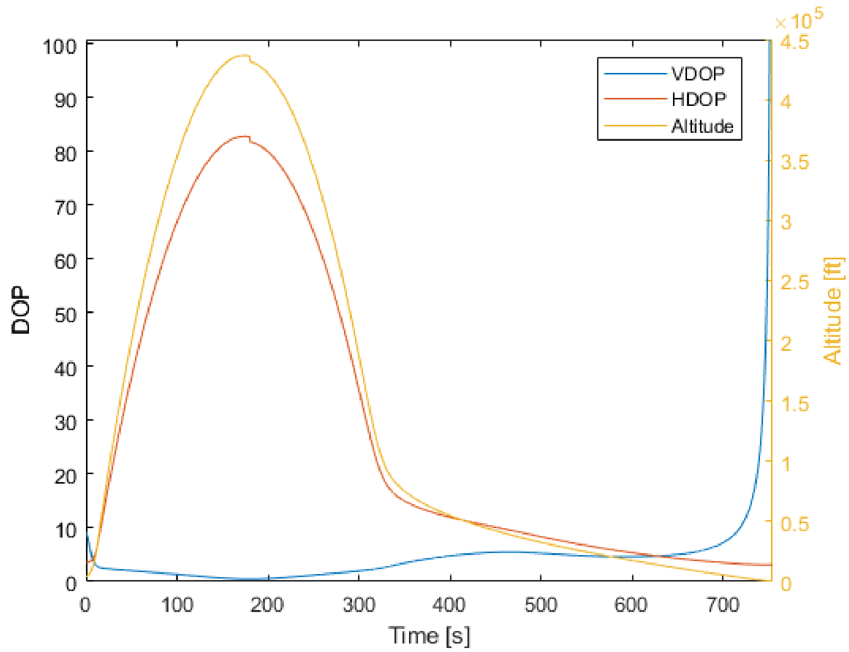


Figure 5.2. Vertical and horizontal DOP plotted for a simulated flight.

This simulation showed that at apogee, the HDOP reached 82.7 and the VDOP was 0.5. Plugging in the known standard deviation of the rangefinder from Test 6, this produces a standard deviation on position of 4102 m in the horizontal plane and 25 m in the vertical plane. While the horizontal error might seem excessive, this is okay – the vertical position is what we care about in this situation, and this is extremely accurate at apogee! Interestingly, during descent, the roles reverse: HDOP enters the single digits as the rocket descends, while VDOP begins to asymptotically approach infinity as the rocket moves into the same plane as the ground stations. This effect should be very useful for determining the rocket's landing position, where the data of interest is the rocket's horizontal landing site on the surface of the ground. Based on these results, we can definitively say that the error behavior of the rangefinder is very favorable for rocketry applications!

Deducing Position: The Trilateration Algorithm

While we can now estimate error, the core problem of the rangefinder still remains: converting distances recorded by each ground station into a single point of estimated position. As discussed in Chapter 2, the standard way to solve this is with a nonlinear minimum finding algorithm. First, based on ranges from the ground stations, an error function $E(\vec{p})$ is created that describes how close each point $\vec{p}$ is to an ideal solution. Then, a minimum finding algorithm is used to locate the point in 3D space that produces the lowest value of this function. This minimum point provides the best estimation of the rocket's range at this point in time.

To test this method, we implemented our own trilaterator program in in C++ for performing this computation process. Because some RPL members had familiarity implementing these computation methods, we elected to implement our own nonlinear solver (though public implementations do exist, such as Eigen::LevenbergMarquardt). Initially, the Gauss-Newton method was used, which operates as effectively a 3D analogue of Newton's Method for iteratively finding zeros of a function. However, this method didn't converge well in our tests for the rangefinding problem, so we switched to the more complex Levenberg-Marquardt algorithm. This algorithm intelligently interpolates between the Gauss-Newton method and the alternative gradient descent method, which calculates the gradient vector at a point, then iteratively moves in the direction of negative gradient. We also made use of the shift-cutting method, which decreases the size of the current Gauss-Newton step and tries again with a smaller step if the step would cause the error function to increase (it is traveling "uphill"). This revised algorithm proved effective at solving trilateration systems, and we have not had any more issues with poor convergence.

To use these algorithms, an initial guess on position is needed. The more accurate the guess is, the less computation work is needed to reach a solution. Since physics requires the rocket to move in a continuous line, its position at the previous point is a very good starting point for trilateration of the next point. So, when processing a continuous line of points, the trilaterator is able to iterate very quickly: in our testing, we were able to process a 90,000 point trajectory in less than a second. However, in real systems, large gaps in the trajectory could occur if the rangefinder loses connection for a portion of the flight. The program's behavior under these conditions has not yet been characterized.

To test the trilaterator, we performed a similar procedure as in the previous section using a simulated flight of RPL's next space shot. First, position data was fed through a simulation of the rangefinder where, at each point, the distance to each ground station was calculated and output. Three ground stations were used placed in an equilateral triangle, with each ground station 4 mi (6.4 km) from the launch site (roughly the distance used in Test 6). After distances were taken, to simulate measurement error in the rangefinder, random error was added according to a uniform distribution with a radius of 100 m. Then, the flight data was processed through the aforementioned trilaterator algorithm. Finally, it was plotted, both in 1D plots and as a 3D trajectory (Figures 5.3 through 5.8).

However, these plots formed a fairly difficult-to-parse point cloud that could be difficult to analyze, especially in the high-error horizontal axes. To smooth out these plots into a single line, we attempted running a LOESS regression on the data in each axis. LOESS works by fitting a linear regression at a number of different places in the data (we used 50 different regressions for 90,000 points of data). This regression is locally weighted: points closer to the center are given more weight than points further away. This makes LOESS a good fit (so to speak) for approximating a number of data points into a continuous but arbitrary curve, such as the aerodynamic trajectory of a rocket's flight [13]. So, we found and integrated an open-source library [14], tested several different span configurations until we found the one that produced the closest fit, and added it to the plots.

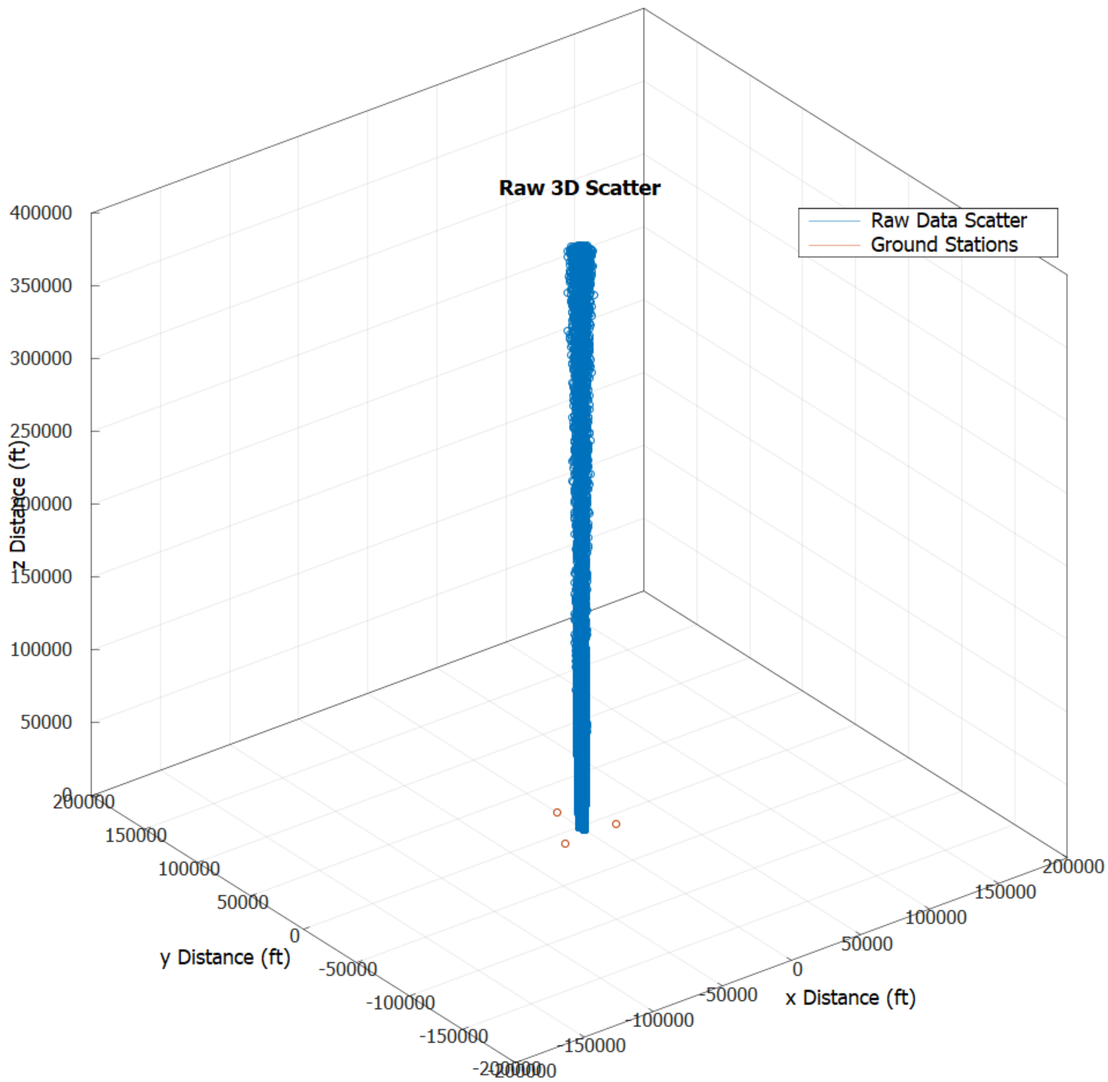


Figure 5.3. Simulated ranging of a spaceshot. Three-dimensional scatter plot of trilaterated points. All axes have the same scale in this plot. Note that upwards and downwards trajectories are so close together they can't be distinguished.

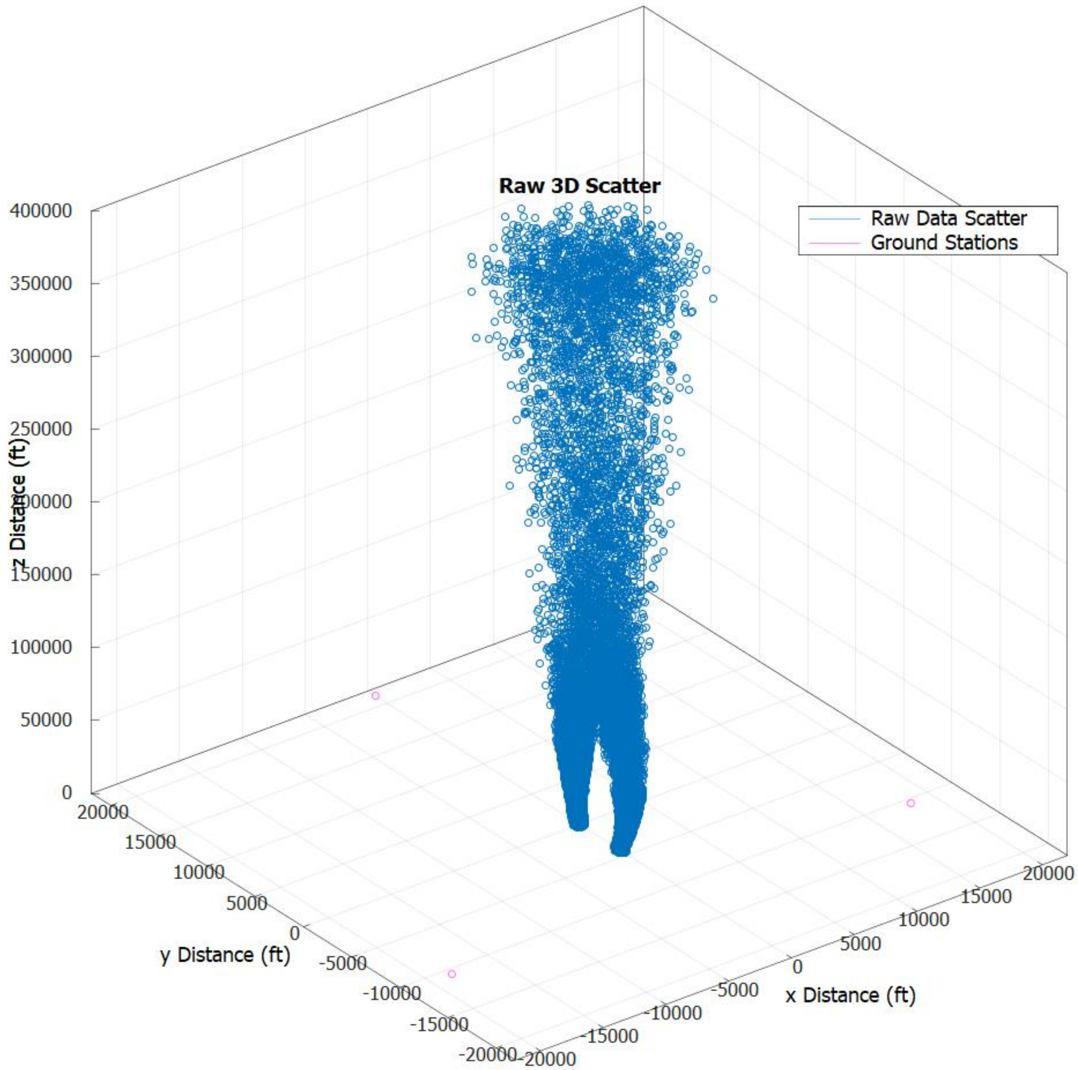


Figure 5.4. Same as previous figure except X and Y axes have been scaled up 10x to show horizontal error in the data points.

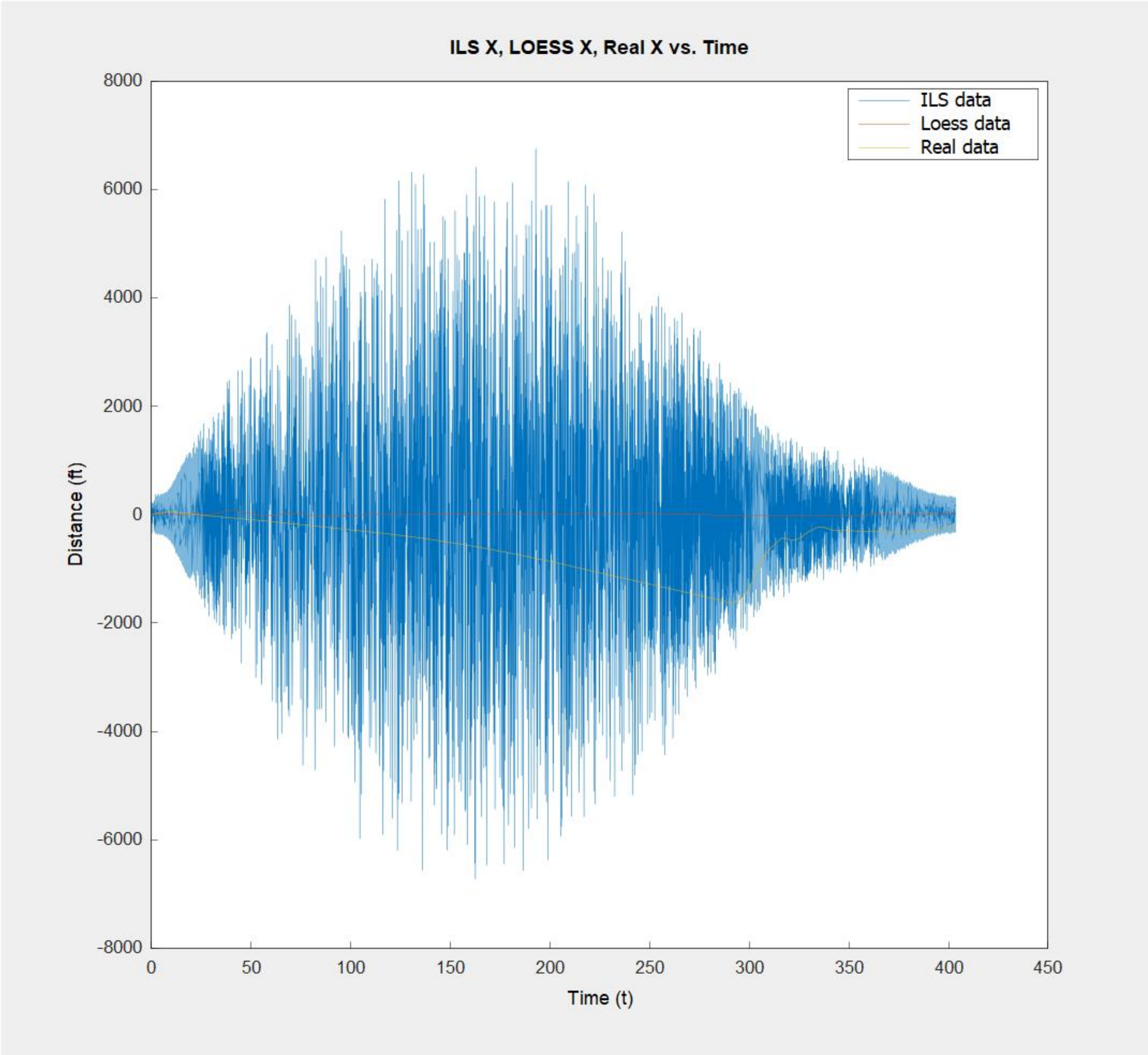


Figure 5.5. Simulated ranging of a spaceshot. X-axis scatter plot of trilaterated points compared with real data and the LOESS regression.

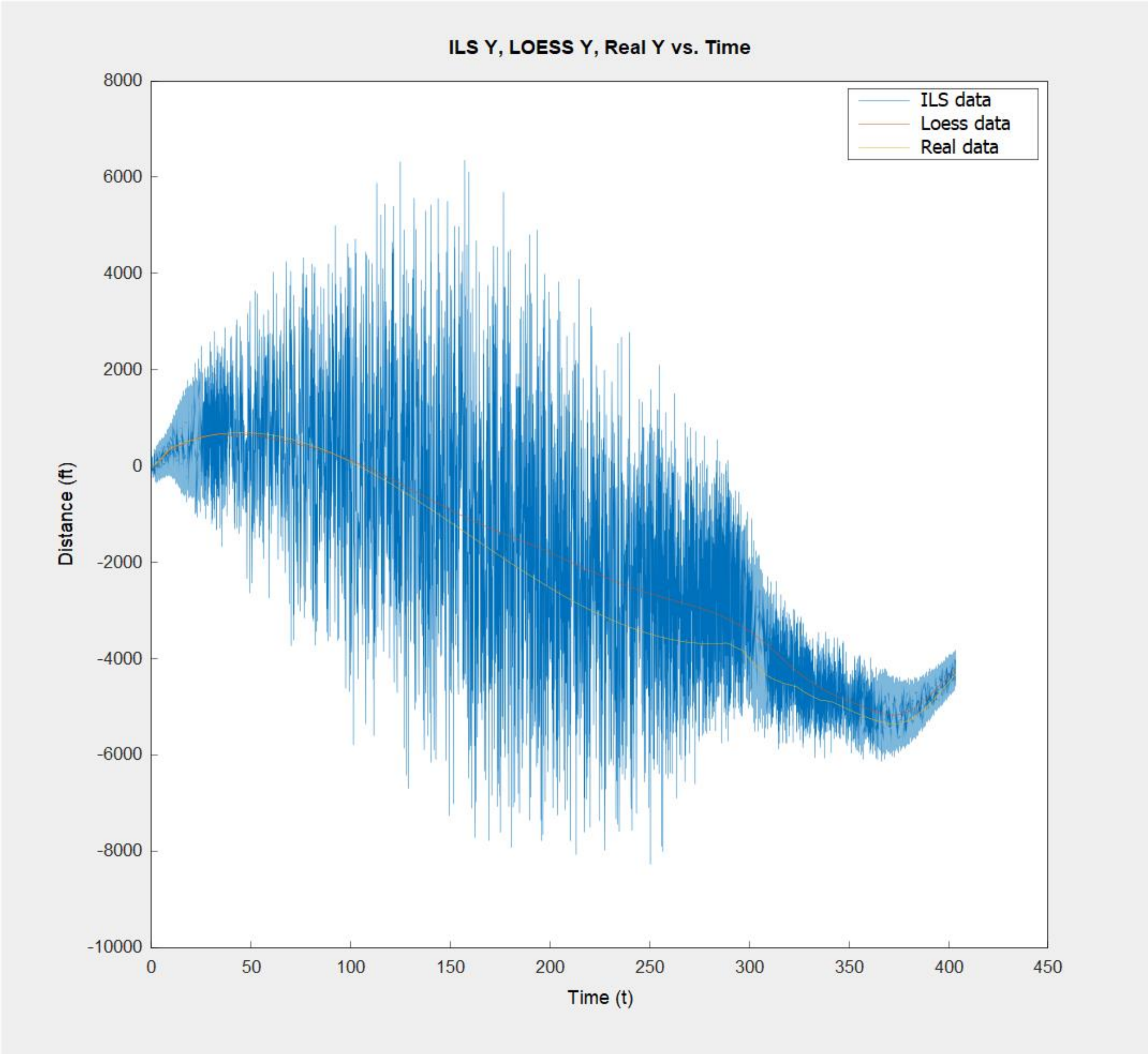


Figure 5.6. Simulated ranging of a spaceshot. Y-axis scatter plot of trilaterated points compared with real data and the LOESS regression.

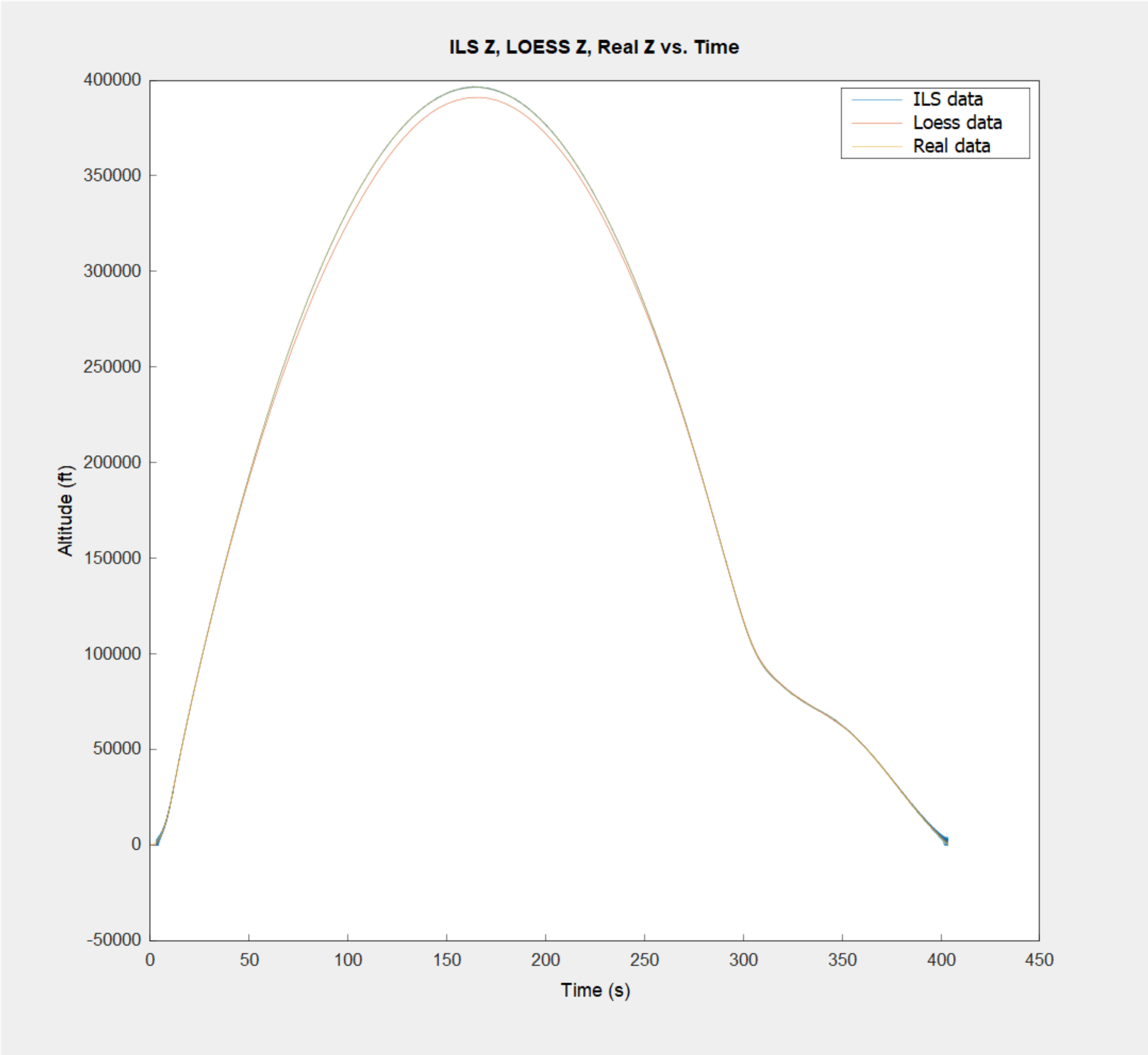


Figure 5.7. Simulated ranging of a spaceshot. Z-axis scatter plot of trilaterated points compared with real data and the LOESS regression.

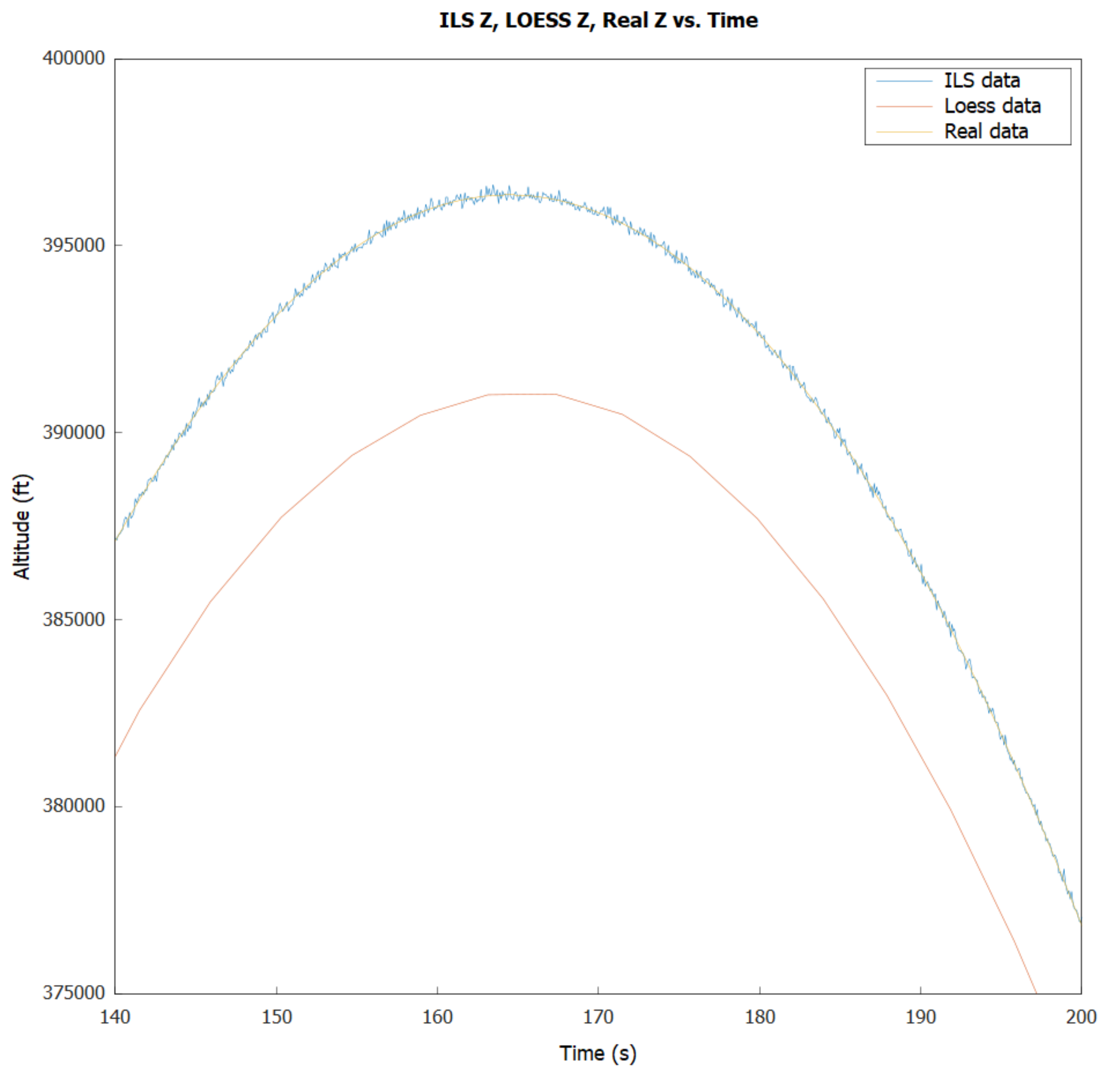


Figure 5.8. Glamour shot showing a close-up of apogee on the previous plot.

These graphs indicate that the Levenberg-Marquardt trilaterator performed admirably, providing a scatter plot that visibly centers on the real curve. Horizontal uncertainty is huge once the rocket reaches a high altitude, but this is expected and doesn't have a serious impact on determining apogee. As figure 5.8 shows, apogee itself was correctly predicted to within a few hundred feet even without any averaging or regression of the trilaterated points.

However, the LOESS regression did not perform quite as well. On the Z-axis plots it underpredicted apogee by thousands of feet, likely because it averaged a large number of points before and after the time of apogee. To remedy this, we could look into changing its configuration to use quadratic regressions or to use a smaller window of data for each point. However, it also exhibited very strange behavior on the X and Y plots, straying thousands of feet off from the correct value at certain points. We are unsure if this is simply a bug in our code or the library that we haven't been able to track down, or if this is a manifestation of LOESS regression's poor ability to handle outliers in the data set [13].

In the future, we will likely continue to use the same trilateration method, but we may switch to a different method of producing a smoothed curve if we cannot resolve the issues seen with LOESS regression. Potentially a simple average could be used, or alternate statistical methods such as a non-linear least squares regression. For determining altitude at least, the data is already accurate enough that only a very light statistical hand should to be needed to create a smooth, accurate curve.

Conclusion

Over the past year and a half, USC RPL has successfully created a fairly novel method of tracking a rocket's flight path and apogee. The Lightspeed Rangefinder has demonstrated correct operation over a distance of 6.8km and, based on our calculations, should easily operate across over a hundred kilometers once the final version is complete. And while its absolute accuracy varies based on placement of the ground stations, position of the vehicle, and the coordinate axis considered, we are confident that the rangefinder will be accurate enough for determining both the apogee and landing location of high power rockets. We believe that this system will be extremely useful in the future both for USC RPL, and for other amateur rocketry groups who wish to accurately measure their apogees.

In this paper, we attempted to show not just the results achieved with the rangefinder, but also the storied process of designing and testing the system. Our hope is that this will illustrate both the struggles and the triumphs involved in constructing a complete embedded radio system, and aid other groups who are following the same path towards testing and verifying a digital radio. As the problem of measuring apogee accurately is one faced by many rocketry groups, USC RPL is open to assisting other teams that wish to build a similar system. The core radio schematics are discussed in Chapter 3, and much of the radio-related code has been released in Appendix A. While we are not currently releasing the remainder of the design, such as the data logging and timing code, our hope is that the pieces we have released will take the "guesswork" out of creating one's own rangefinder. The rest, as they say, is just embedded systems work!

USC RPL is also open to loaning out our rangefinder system to other amateur groups planning high-altitude launches who desire confirmation of their trajectory and apogee. Those interested, as well as anyone with questions about how the rangefinder works, should contact flighton@uscrpl.com for details.

Now that we have reached the milestone of completing initial qualification tests on the system and solidifying a final design, USC RPL will continue preparing the rangefinder for usage on a space shot. This will include more stringent in-lab qualification tests, a redesigned and upgraded transponder PCB, and (of course) further attempts at an in-flight qualification test on an amateur rocket (or even a plane). The Lightspeed Rangefinder is off to a promising start, and we believe that it will be completely ready soon.

Appendix A: Code Releases

The Mbed OS driver written to control the CC1200 radio was released as open source here: https://os.mbed.com/users/MultipleMonomials/code/CC1200/

The code written to transmit Morse code on the CC1200 was released as open source here: https://os.mbed.com/users/MultipleMonomials/code/CC1200-MorseEncoder/

The modified version of the open-source code used in the analysis to perform a LOESS regression has been posted here:

https://github.com/multiplemonomials/loess

Code used to implement Tests 1, 2, and 3, as well as configure radio settings, was released here by RPL:

https://github.com/USCRPL/LightspeedRangefinder-TestingCode

Appendix B: Future Improvements

This section collects a couple of notes on changes we intend to make to the rangefinder system going forward, but that we have not been able to try out yet. Make sure to read through this so you don't make the same mistakes we did!

Improvement	**Details**
Upgraded sensitivity test setup	During lab testing, we ran into issues with RF leakage through the foil box at higher power margins – this prevented us from doing a complete verification of the link budget in the lab. To remedy this, we intend to purchase an RF test chamber (such as the Aeroflex 4921) that provides a much higher level of isolation between the inside and the outside. Additionally, it was pointed out during review of this paper that the attenuation setup we used during testing was not ideal: the Telonic variable attenuator lacks documentation and may not be rated to properly handle the frequencies and power levels used in our tests. While we did check its calibration ourselves, we can't be sure it's behaving exactly the same under high-power test conditions. So, we plan to purchase two high-power (5W) attenuators to sit on either side of the Telonic attenuator and reduce the power coming into it to a much lower value. We might also look into alternate attenuator models in the future if we can find one for an affordable price.
Adding a SAW filter	Another concern brought to us during review was the potential for other rocket-mounted radio transmitters in the 430MHz band to interfere with the transponder's operation. The CC1200 only includes a limited amount of blocking – for example, about 61dB at ±10MHz for our radio settings. This means that if another signal is being transmitted 10MHz away from the rangefinder's signal, but it enters the antenna as more than 61dBm stronger than the received signal from the ground station, it could potentially override the weaker signal and prevent the ground station's transmission from being received. This type of interference could be a serious issue

	going forwards, as RPL is planning to soon add other radios that make use of the 430MHz band and will be continuously transmitting at high power. To counteract this, we plan to add a Surface Acoustic Wave (SAW) filter to the receive input of the transponder. This IC acts as a precise notch filter and admits signals in the 432-435MHz band with 2dB or less of attenuation, but blocks signals below 425MHz and above 445MHz with >40dB of attenuation [15]. By running the rangefinder in the passband and other radios outside of it, and by mounting the antennas perpendicularly to each other, we should be able to obtain very good isolation between the rangefinder and other radio systems.
Ionospheric delay analysis	As one reaches the upper levels of the atmosphere, the ionosphere becomes more dense and begins to have a noticeable effect on radio signals. In real GPS systems, corrections have to be applied because radio signals propagate slightly slower through the ionosphere than through free air, which would introduce error to range measurements based on those signals. Since the rangefinder works off of the same principle, this error could affect it as well. Some back-of-the-envelope calculations showed that ionospheric delay would be pretty much insignificant up to 100 km of altitude, but as we work towards higher altitudes and increased accuracy, we should investigate this phenomenon further and determine if we need to adjust for it in our analysis.
Looking into dropped packets	Many of our tests, such as Test 6, observed a relatively high number (~70%) of dropped packets despite relatively good signal conditions. When this was occurring, ranging still worked correctly, but the effective sampling rate was reduced. This phenomenon was consistently noticed in tests when signal quality dropped below ideal values, but well before the power margin reached close to zero – based on Test 3, under these conditions there shouldn't be a transmission error more than once every few hundred packets. It is

	unknown what is causing this effect, as it was not deemed a priority to debug before launch, but an issue with the software is suspected. Hopefully, we can diagnose the problem and recover our lost packets!

Works Cited

[1] USC Rocket Propulsion Lab, "Traveler IV Apogee Analysis," May 2019. [Online]. Available: http://www.uscrpl.com/s/Traveler-IV-Whitepaper.

[2] TE Connectivity, "MS5607-02BA03," June 2017. [Online]. Available: http://tinyurl.com/f7utsbw8.

[3] Texas Instruments, "CC1200 Datasheet," 29 September 2014. [Online]. Available: http://www.ti.com/lit/gpn/cc1200.

[4] W. Navidi, W. S. Murphy and W. Hereman, "Statistical methods in surveying by trilateration," *Computational Statistics and Data Analysis,* vol. 27, no. 2, pp. 209-227, 1998.

[5] Texas Instruments, "CC1200 User's Guide," 26 September 2013. [Online]. Available: http://www.ti.com/lit/pdf/swru346.

[6] F. T. Ulaby and U. Ravaioli, Fundamentals of Applied Electromagnetics, Upper Saddle River, New Jersey: Pearson Education Inc., 2015.

[7] Texas Instruments, "TPS562207 4.3-V to 17-V Input, 2-A Synchronous Buck Converter in SOT563," April 2021. [Online]. Available: https://www.ti.com/lit/gpn/tps562207. [Accessed 17 April 2021].

[8] F. Baltieri, "LNA for RTL Based SDR Receivers," 22 June 2014. [Online]. Available: https://fabiobaltieri.com/2014/06/22/lna/. [Accessed 18 April 2021].

[9] Nordic Semiconductor, "RF Performance Test Guidelines White Paper," Novenber 2008. [Online]. Available: https://infocenter.nordicsemi.com/pdf/nwp_006.pdf.

[10] Skyworks Inc., "SKY65366-21: 400 MHz Transmit/Receive Front-End Module," 7 June 2018. [Online]. Available: https://www.skyworksinc.com/-/media/SkyWorks/Documents/Products/2801-2900/SKY65366_21_203146E.pdf.

[11] W. Tang, X. Ma, J. Wei and Z. Wang, "Measurement and Analysis of Near-Ground Propagation Models under Different Terrains for Wireless Sensor Networks," *Sensors,* vol. 19, no. 8, p. 1901, 2019.

[12] R. Santerre and S. Banville, "Geometry of GPS dilution of precision: revisited," *GPS Solutions,* vol. 21, no. 4, pp. 1747-1763, 2017.

[13] NIST, "LOESS (aka LOWESS)," 30 October 2013. [Online]. Available: https://www.itl.nist.gov/div898/handbook/pmd/section1/pmd144.htm. [Accessed 2 May 2021].

[14] B. Vermeulen, "loess," 27 August 2018. [Online]. Available: https://github.com/bartverm/loess.

[15] Qualcomm RF360 Europe GmbH, "B39431B3710U410 Data sheet," 2019.