
by

Ian Graham Bennett

A thesis submitted in conformity with the requirements for the degree of Master of Applied Science
Graduate Department of Aerospace Science and Engineering
University of Toronto

© Copyright 2017 by Ian Graham Bennett
Abstract


Ian Graham Bennett
Master of Applied Science
Graduate Department of Aerospace Science and Engineering
University of Toronto
2017

Automatic Dependent Surveillance-Broadcast (ADS-B) is quickly becoming the new standard for more efficient air traffic control, but as a satellite/ground-based hybrid system it faces limitations on its usefulness over oceans and remote areas. Tracking of aircraft from space presents many challenges that if overcome will greatly increase the safety and efficiency of commercial air travel in these areas. This thesis presents work performed to develop a flight-ready ADS-B receiver payload for the CanX-7 technology demonstration satellite. Work presented includes a simulation of payload performance and coverage area, the design and testing of a single-feed circularly polarized L-band antenna, the design of software to control the payload and manage its data, and verification of the performance of the hardware prior to integration with the satellite and launch. Also included is a short overview of results from the seven-month aircraft tracking campaign conducted with the spacecraft.
Acknowledgements

My time as a graduate student at SFL has been both one of the most challenging and rewarding times of my life. I’d like to thank my supervisor Dr. Robert Zee for giving me the opportunity to come to the University of Toronto and be a part of such a fantastic group at SFL. The SFL students and staff have been amazing, it’s been great to work with so many insightful, knowledgeable people over the past three years on some really cool projects. I’d also like to acknowledge my fellow students Paris Ang, for instructing me in the nuances of HFSS and for working with me on the CanX-7 antenna, and Clyde Wheeler, for answering all my Python questions. I’d also like to thank my project managers Grant Bonin and Brad Cotten for giving me the opportunity to work on such a unique project; it’s been a pleasure working for you (even when I had to fold drag sails).

I’d also like to acknowledge the friends and family who’ve been with me along the way. Brenan, Eric, Jeff, Rob, and Stan, thanks for always being a welcome distraction from whatever it is that I probably should be doing. To all the SFL staff and students, it’s been fantastic working with you, and even more fantastic getting to know you. To my aunt Martha, thanks for helping me get on my feet (and stay on them) after moving to Toronto. The help you’ve given me over the past three years has been amazing, I couldn’t have finished this without you.

Lastly, I’d like to thank my parents and my brother Alex for always being there when I’ve needed you, and for making everything I’ve accomplished possible so far. You’ve been more patient and understanding than I deserve; thanks for believing in me.
Contents

Acknowledgements iii

Table of Contents vi

List of Figures viii

1 Introduction 1

1.1 Aircraft Traffic Management ........................................ 1
  1.1.1 ADS-B and Cooperative Surveillance .............................. 2

1.2 Previous ADS-B Detection Missions ................................. 3
  1.2.1 Proba-V Secondary Payload ........................................ 3
  1.2.2 GOMX-1 and GOMX-3 ............................................. 4

1.3 Previous SFL Missions .............................................. 4
  1.3.1 AISSat Constellation ............................................. 5
  1.3.2 Norsat-1 and Norsat-2 ........................................... 6

1.4 CanX-7 .......................................................... 6
  1.4.1 Spacecraft Bus ................................................... 7
  1.4.2 Deployable Drag Sails ............................................ 8
  1.4.3 ADS-B Receiver Secondary Payload ............................... 9

1.5 Thesis Scope and Motivation ....................................... 9

2 Air Traffic Control Systems and the CanX-7 ADS-B Payload 12

2.1 Traditional Air Traffic Surveillance Techniques ................... 12
  2.1.1 Primary Surveillance Radar ...................................... 12
  2.1.2 Secondary Surveillance Radar .................................... 14
  2.1.3 Primary and Secondary Surveillance Radar Coverage .......... 15

2.2 Automatic Dependant Surveillance-Broadcast ....................... 15
  2.2.1 Aircraft Transmitter Hardware and Regulations ............... 16
  2.2.2 ADS-B Physical Layer .......................................... 17
  2.2.3 ADS-B Data Link Layer ......................................... 18

2.3 CanX-7 Payload Hardware Configuration ............................ 19
  2.3.1 Royal Military College of Canada Prototype ................... 19
  2.3.2 Improvements to Prototype Payload ............................ 19

2.4 CanX-7 Payload Concept of Operations ............................ 21
### 3 Link Analysis Background

- **3.1 Transmitter**
  - 3.1.1 Transmitter Power
  - 3.1.2 Bit Rate and Energy Density
  - 3.1.3 Transmitting Antenna

- **3.2 Environment and Geometry Effects**
  - 3.2.1 Free-Space Path Loss
  - 3.2.2 Atmospheric Loss

- **3.3 Receiver**
  - 3.3.1 Receive Antenna
  - 3.3.2 Receiver Electronics

- **3.4 Link Margin Calculation**
  - 3.4.1 Bit Energy to Noise Power Spectral Density Ratio

### 4 Coverage Simulation

- **4.1 Single Position Coverage Simulation**
  - 4.1.1 Link Equation Positionally Independent Terms
  - 4.1.2 Link Equation Positionally Dependent Terms
  - 4.1.3 Spacecraft Antenna Gain
  - 4.1.4 Message Decoding Percentage

- **4.2 Single Spacecraft Position Results**

- **4.3 Coverage Swath**
  - 4.3.1 Coverage Swath Results

- **4.4 Conclusion**

### 5 Antenna Design and Testing

- **5.1 Modification of the Prototype Antenna Design**
  - 5.1.1 Patch vs. Monopole Antenna

- **5.2 Patch Antenna Design**
  - 5.2.1 Antenna Design Simulation
  - 5.2.2 Initial Design Manufacturing
  - 5.2.3 Redesign Using Design Dielectric Constant

- **5.3 Antenna Pattern Testing**
  - 5.3.1 Setup
  - 5.3.2 Test Plan
  - 5.3.3 Results

- **5.4 Conclusion**

### 6 Payload Control Software

- **6.1 Prototype Software and Requirements Generation**
  - 6.1.1 Software Requirements Generation
  - 6.1.2 Software Architecture

- **6.2 Software Development Testbed**

- **6.3 Payload Computer Boot Sequence**
List of Figures

1.1 Subset of aircraft observed by the Proba-V AOS payload [22] .......................... 3
1.2 Rendering of the GOMX-1 spacecraft with ADS-B antenna deployed [9] ............. 4
1.3 The AISSat-1 spacecraft bus ............................................................................. 5
1.4 The Norsat-1 (left) and Norsat-2 (right) spacecraft ........................................... 6
1.5 The CanX-7 spacecraft, with deployed magnetometer/camera boom and UHF antennas .................................................................................. 7
1.6 CanX-7 nominal attitude profile ....................................................................... 8
1.7 The CanX-7 spacecraft with drag sails stowed (left) and deployed (right) ........... 9
1.8 Exploded view of the ADS-B payload ................................................................. 10

2.1 Illustration of the PSR technique ...................................................................... 13
2.2 Illustration of the SSR technique ...................................................................... 14
2.3 Primary (left) and Secondary (right) Radar Coverage In Canada [38] ................. 15
2.4 ADS-B baseband binary pulse-position modulation [34] ................................. 18
2.5 Breakdown of ADS-B frame data ....................................................................... 18
2.6 ADS-B payload prototype electronics integrated for dirty-sat assembly ............ 20
2.7 Diagram of modifications to the ADS-B payload avionics ............................... 21
2.8 CanX-7’s ADS-B payload antenna ..................................................................... 22

3.1 ADS-B Uplink Budget .................................................................................... 24
3.2 An unmodulated carrier signal (left) and one modulated at 1 MHz using PPM (right) . 25
3.3 Diagram of an electromagnetic wave [47] ......................................................... 27
3.4 An example polarization ellipse ....................................................................... 28
3.5 Atmospheric loss due to oxygen gas and water vapor [46] ............................... 32
3.6 The CanX-7 ADS-B receiver’s receive chain ................................................... 35

4.1 Angles defining the position of aircraft in the simulation ................................. 44
4.2 Problem geometry used to calculate aircraft antenna gain and environment losses 46
4.3 Coverage with the spacecraft at various true anomaly values (clockwise, from top-left): 0°, 20°, 80°, 40° ................................................................. 49
4.4 Coverage swath during a typical 18 minute recording pass (left) and over a full spacecraft orbit (right) ................................................................. 50

5.1 The prototype antenna (left), next to the final SFL designed antenna (right) ....... 51
5.2 Line of sight coverage of the Earth’s surface for monopole (left) and cardioid patch (right) antennas, with the spacecraft located at (0,0) ................................. 53
Chapter 1

Introduction

In recent years, interest has been growing in establishing a cost-effective and reliable Air Traffic Control (ATC) system that can implement real time global tracking of civilian aircraft [35]. The Automatic Dependent Surveillance-Broadcast (ADS-B) system has been around for more than 15 years, and was originally conceived as a replacement for the current primary and secondary radar-based ATC network [24]. With ADS-B, rather than requiring a fixed ground-based transmitter to determine aircraft positions, the aircraft determine their own position through a combination of Global Navigation Satellite System (GNSS) measurements and other instrumentation. They then broadcast messages at semi-random intervals containing their position and flight information that can be received by aircraft in the vicinity.

With many countries around the world implementing regulations requiring the use of ADS-B on commercial aircraft, this system is already prevalent around the world, and will only grow more so with time. Since the ADS-B system does not require interrogation from the ground or a primary radar to determine aircraft positions, it is an ideal system to be implemented via satellite detection. Beginning in 2018, a space-based ADS-B monitoring system operated by Aireon and hosted on the Iridium-NEXT constellation of satellites will begin monitoring aircraft activity across the world [21]. These payloads are, however, very expensive and reliant on a host constellation to operate. A small satellite constellation using a cheaper but still flight-proven receiver could potentially achieve similar results at a fraction of the cost.

This thesis presents the design and testing of an ADS-B receiver payload flown on the Canadian CanX-7 spacecraft by the University of Toronto Institute for Aerospace Studies Space Flight Lab (UTIAS-SFL). Discussed are background information on ADS-B and link analysis techniques, the design of the receiver, and the payload mission objectives. Also discussed are simulations performed to estimate the receiver performance, design work performed on the payload’s antenna, software written to control and manage the payload, and the testing methods and results used to ensure that the payload would perform properly once on-orbit.

1.1 Aircraft Traffic Management

Reliable aircraft surveillance is an important part of a successful air traffic control system. By monitoring the locations of aircraft along their flight paths, controllers maintain adequate spacing between aircraft,
and ensure that collisions between them are avoided. In areas with inadequate tracking and surveillance, aircraft flight paths must be spaced widely apart, leading to inefficiencies that can substantially increase flight times and fuel costs. With the advent of Global Navigation Satellite Systems (GNSS) for position determination, cooperative air traffic surveillance techniques have become more feasible, and have the potential to supplement and bridge the gaps between areas covered by traditional surveillance radar, with the goal of eventually supplanting it entirely. With accurate, global, real-time surveillance, aircraft can be spaced closer together on their flight paths, inefficiencies can be eliminated, and passenger safety can be improved.

1.1.1 ADS-B and Cooperative Surveillance

A newer, cooperative aircraft surveillance standard developed in the early 2000’s, Automatic Dependent Surveillance-Broadcast (ADS-B) is designed to leverage high-precision GNSS positioning to supplement and replace primary and secondary surveillance radar (SSR). ADS-B operates on the same frequency and with the same modulation scheme as the Traffic Collision Avoidance System’s (TCAS) response messages, but has an extended message length called Extended Squitter. In addition to containing an aircraft’s International Civil Aviation Organization (ICAO) identification number, as TCAS replies do, an ADS-B message contains additional information about the aircraft, and is broadcast automatically instead of having to be requested by a ground station. This information can be used by ground controllers to identify and locate aircraft without much of the costly infrastructure required by traditional techniques.

While less expensive to install and maintain than primary or secondary radar, the current ADS-B monitoring network’s ground-based infrastructure still limits the system’s surveillance capabilities to aircraft flying over land. It is not practical to place receivers at sea, and it can be expensive to extend the needed data and maintenance services to remote areas such as the poles or deserts. Space-based ADS-B offers many advantages over a ground-based system, as spacecraft are not limited to covering any particular area of the globe, and a receiver in space has a larger coverage area than a ground-based receiver, which has its range limited by line-of-sight.

For space-based ADS-B to work, a functional system requires a constellation of multiple satellites in order to obtain constant real-time global coverage. In 2018, Aireon LLC will become the first company to obtain global real-time ATC coverage, with an ADS-B receiver hosted on each of the Iridium NEXT communication satellites [21]. The price tag of the Iridium NEXT mission is $2.3 billion US, not including launch costs, and its primary mission is global communication; the ADS-B receiver is a hosted secondary payload.

By using a smaller, dedicated spacecraft to host a more compact payload, a constellation of ADS-B monitoring satellites could be built at much lower cost. A dedicated small spacecraft constellation has several advantages over a hosted payload solution. It is less expensive to build spare satellites, allowing for redundancy in the case of a failure in one of the spacecraft, and the constellation can be more responsive to technological and regulatory changes, with quicker development and manufacturing times. A small spacecraft constellation can also be built for a fraction of the cost of a large one.

In the interest of developing the technology required for such a small satellite constellation, the University of Toronto Institute for Aerospace Studies Space Flight Lab has developed a compact ADS-B receiver payload in partnership with the Royal Military College of Canada for the CanX-7 technology demonstration spacecraft, in order to validate technologies that could be used in such a constellation.
1.2 Previous ADS-B Detection Missions

The CanX-7 spacecraft is one of the first missions to demonstrate ADS-B detection from space, but there are other missions that have already proven that the concept is a feasible one. Proba-V, a European Space Agency (ESA) mission to map Earth’s vegetation, carried a receiver along with a high-gain antenna to first demonstrate ADS-B reception from orbit. GOMX-1 and GOMX-3 both proved that the same thing could be done with a far smaller spacecraft.

1.2.1 Proba-V Secondary Payload

The German Aerospace Center (DLR) began investigating the possibility of receiving ADS-B messages from space in 2008, with the creation of its ADS-B over Satellite (AOS) project [22]. This group worked on the design of a space-based ADS-B receiver to be included as a secondary payload on the Proba-V mission. This culminated in the launch of the Proba-V spacecraft on May 7, 2013. The Proba-V AOS payload used a two-element patch antenna array in combination with a superheterodyne receiver [22] to detect ADS-B messages. Figure 1.1 shows a small subset of aircraft positions detected by the receiver, along with the satellite’s orbital track.

![Image of aircraft positions detected by the Proba-V AOS payload](image.jpg)

Figure 1.1: Subset of aircraft observed by the Proba-V AOS payload [22]

A superheterodyne receiver uses a mixer to mix a received signal down from its carrier frequency to another fixed lower frequency before demodulating it. The biggest advantage to the superheterodyne receiver is that by varying the local oscillator (LO) frequency that the input signal is mixed with, the receiver can be more selective to a specific channel defined by fixed frequency filtering and amplification on the resulting intermediate frequency (IF), while still being able to electronically vary the center frequency of the front end [6]. The Proba-V AOS payload uses a multi-stage chain of filtering and amplification to condition the received carrier frequency before mixing it down from 1090 MHz to 70 MHz, after which the IF is amplified and filtered again before analog to digital conversion [22].

The antenna used on the AOS payload is a two element patch array, with a peak gain of 11.2 dBi [22]. Patch antennas are a subtype of microstrip antennas, first described in 1972 by J. W. Howell [29], that are frequently used in space applications because of their moderate gain performance and flexibility of
form factor during the design process. They can be made in various shapes and sizes for a wide variety of frequency ranges through the use of various dielectric materials. Multiple patch elements can also be used to create a patch array, which can be used to increase the gain in a specific direction at the cost of overall beamwidth. The antenna used on the Proba-V payload is a two-element patch, fed so as to be right hand circularly polarized (RHCP). It is manufactured without a dielectric material so as to increase the gain, at the cost of increased overall size.

1.2.2 GOMX-1 and GOMX-3

Shortly after the launch of Proba-V, the GOMX-1 satellite was launched on November 21, 2013. GOMX-1 was built by GOMspace, a company located in Denmark specializing in small satellites, and was intended to demonstrate new bus avionics systems and a software-defined radio (SDR) based ADS-B receiver [9]. The spacecraft was a 2U cubesat measuring 20 x 10 x 10 cm, with a long, helical L-band antenna extending from one of the small faces, shown in Figure 1.2. The GOMX-1 mission proved that ADS-B could be detected from space using a small spacecraft for the first time.

![Figure 1.2: Rendering of the GOMX-1 spacecraft with ADS-B antenna deployed [9]](image)

The GOMX-3 mission was similar to GOMX-1, but incorporated a number of improvements made to the original satellite design as well as a number of additional technology demonstrations. The satellite demonstrated new attitude control techniques used to more accurately point the ADS-B antenna at locations of interest, and also hosted an X-band high data-rate transmitter. The mission began on October 5, 2015 when the spacecraft was launched from the International Space Station (ISS) into a 400 km orbit [2]. During the life of the mission, ADS-B data was collected using the receiver and the various other components of the spacecraft were demonstrated. Of note was the near-real time data collection performed by the spacecraft during one demonstration. The satellite was configured to detect aircraft from over the ocean and then downlink their positions to a ground station with less than five minutes passing between the events [4].

1.3 Previous SFL Missions

While SFL has never before flown an ADS-B receiver, several missions have flown Automatic Identification System (AIS) receivers for ship tracking. AIS is an automated system of cooperatively surveilling
ships, with a similar concept of operations as ADS-B, although the implementation is slightly different. Like with ADS-B, ships use information derived from GNSS to accurately position themselves, and then broadcast the information out to be received by other ships and land-based receiving stations in the vicinity. ADS-B detection does provide some distinct challenges from AIS though, in particular the higher operating frequency, higher data rate, and lack of channel access coordination between transmitters to reduce the frequency of message collisions. Despite these differences, much of the experience gained building AIS receiving satellites can be applied towards ADS-B. SFL has so far built seven AIS satellites, with six operating on-orbit, one currently awaiting launch near the end of 2017, and several more under development [7].

1.3.1 AISSat Constellation

The AISSat Constellation consists of three satellites built and launched for the Norwegian Defence Research Establishment (FFI) and Norwegian Space Centre: AISSat-1 and AISSat-2, currently operating on orbit, and AISSat-3, currently awaiting launch at the end of 2017 [7].

The primary mission objective of AISSat-1 is to provide enhanced surveillance of ships over Norwegian territorial waters [13]. The satellite is built on the SFL Generic Nanosatellite Bus (GNB), with a form factor of 20 x 20 x 20 cm, with a magnetometer boom and a fixed VHF payload antenna extending from two of the faces. The spacecraft uses an S-band transmitter and Ultra High Frequency (UHF) receiver for telemetry and command, a combination of sun sensors, a magnetometer, reaction wheels, and magnetorquers for attitude determination and control, and a GPS receiver for on-board orbit determination [13]. An image of AISSat-1 is shown in Figure 1.3.

![Figure 1.3: The AISSat-1 spacecraft bus](image)

AISSat-1 was launched on July 12, 2010 into a polar orbit using a Polar Satellite Launch Vehicle (PSLV). The satellite collects data while operating in one of two distinct modes. The first is a simple data record and store mode, whereby AIS information is stored on the spacecraft after reception for later downlink. The second is a real-time mode, whereby AIS information is received by the satellite and downlinked immediately while over an available ground station. This mode provides latency of less than 1 second between message reception by the satellite and transferral of the message to the Norwegian Coastal Authority (NCA) [13].

Due to the success of the AISSat-1 mission, two more satellites were ordered to supplement the first. AISSat-2 and 3 are essentially build-to-print copies of the first AISSat, each with a successively
upgraded AIS receiver payload. The additional AISSats and are intended to reduce the time between
target revisits, increase observational availability, and improve reliability against hardware faults [13].

1.3.2 Norsat-1 and Norsat-2

Building on the success of the AISSat missions, the Norsat-1 and Norsat-2 missions are two Norwegian
satellites designed and built at SFL. Both satellites are based on SFL’s Next-generation Earth Monitoring
and Observation (NEMO) bus, approximately 40 x 30 x 20 cm in size, and include a top-mounted solar
panel wing to increase power generation [1]. The two satellites can be seen in Figure 1.4. Both satellites
contain an AIS receiver, with Norsat-1 also carrying two science instruments, a radiometer and a set of
Langmuir Probes [7], and with Norsat-2 carrying a prototype VHF Data Exchange System (VDES) [14].
Both satellites were launched on July 14, 2017 aboard a Soyuz rocket. Since the launch, both spacecraft
have been commissioned and are operating nominally.

Figure 1.4: The Norsat-1 (left) and Norsat-2 (right) spacecraft

Norsat-1’s AIS mission is similar to the AISSat Constellation, in that it is primarily intended to
increase AIS data flow to the Norwegian Coastal Authority, while improving the uptime and reliability
of the constellation overall. Norsat-2 also serves this same purpose, but in addition aims to demonstrate
wide-area two-way data services to ships crossing the Arctic Ocean [14]. It is also unique in that
it contains a mechanically complex deployable Yagi-Uda antenna as part of its primary payload. The
antenna gives greater directionality than those used on the AISSats and Norsat-1, and allows for increased
receiver sensitivity and transmitter performance with a similar power draw.

1.4 CanX-7

CanX-7 (Canadian Advanced Nanospace eXperiment-7) is a small spacecraft built by the University of
Toronto Space Flight Lab, operating three technology demonstrations:

1. A set of modular, deployable drag sails to demonstrate passive end-of-life deorbiting of small
spacecraft in low Earth orbit (LEO)

2. A compact ADS-B receiver to validate the use of a low-cost, commercial-off-the-shelf (COTS)
ADS-B receiver for space-based aircraft tracking
3. A set of miniature imaging cameras for verifying drag sail deployment and performance

The spacecraft was launched on September 26, 2016 from Satish Dhawan Space Centre in India on a Polar Satellite Launch Vehicle (PSLV). Once contact was made with the spacecraft early the next morning, commissioning took place over the following week with the functionality and performance of each spacecraft subsystem and payload being verified. Once all subsystems and payloads were confirmed to be working, a seven month ADS-B collection campaign was begun. Upon completion of the campaign in late April 2017, the four drag sails were deployed in order to demonstrate their ability to passively de-orbit a small spacecraft. Almost a year after launch, the spacecraft is four months into the de-orbit phase of its mission and is still healthy, sending down telemetry data regularly.

1.4.1 Spacecraft Bus

Measuring 10 x 10 x 34 cm with a mass of 3.6 kg, the CanX-7 spacecraft is a redesign of SFL’s successful CanX-2 bus, and is similar in size and mass to a 3U cube satellite. The CanX-2 bus was chosen for the CanX-7 mission because of its small size. This reduces both launch and build costs, while still providing sufficient space and power for the spacecraft’s payloads. Thermal control is entirely passive, handled via thermal tapes used to keep the spacecraft avionics and payloads within acceptable temperature limits.

![The CanX-7 spacecraft, with deployed magnetometer/camera boom and UHF antennas](image)

The spacecraft is powered by a set of solar panels, with power stored in a lithium-ion battery. CanX-7 is the second SFL spacecraft launched to include the modular power system (MPS), a switched Peak Power Tracking (PPT) power management and distribution system. The power system has several slots in which cards can be inserted, each card providing switched voltage between 3.8 and 5.5 Volts and carrying both hardware and software based overcurrent protection [17].

For telemetry and command, the spacecraft uses a full-duplex Ultra High Frequency (UHF) uplink and S-band downlink communication system, with command and data handling performed by an SFL-designed and built On-Board Computer (OBC). Communications to and from the ground are all handled
through the SFL ground station. A ground station located at SFL uses a directional Yagi antenna transmitting in the UHF band for uplink to the spacecraft, with a maximum power output of 200 Watts. The uplink is modulated at 4 kbps using Gaussian Frequency-Shift Keying. The baud rate of this link is fixed such that it can be decoded in hardware by an SFL designed UHF receiver, and is at a low rate in order to increase the bit energy density and ensure a high link margin in the uplink direction. For the downlink of data and telemetry, CanX-7 uses an SFL-designed transmitter at S-band, capable of transmitting at bit rates between 32 and 1024 kbps, modulated using either Binary or Quadrature Phase Shift Keying (BPSK or QPSK, respectively). This gives the spacecraft the flexibility to operate at either a lower data rate to close a link with the ground station in adverse conditions, or increase the data rate to improve the amount of data that can be transferred in better conditions. A 2.1 meter parabolic dish located at the SFL ground station is used to receive S-band transmissions from the spacecraft.

The spacecraft uses a two-axis attitude control solution with a magnetometer for attitude determination and a set of three magnetorquers for attitude control. Even though this solution doesn’t provide full three-axis control for the spacecraft, it does allow for any vector in the spacecraft frame to be aligned with the local magnetic field, which is sufficient for completing the ADS-B data measurement campaign. Due to the spacecraft’s polar orbit, if the ADS-B antenna boresight is aligned with the local magnetic field lines, it is pointed towards Earth in an optimal direction while the spacecraft is passing over the North Atlantic Ocean, and if the polarity of the magnetorquers are reversed, the antenna is pointed towards Earth while the spacecraft passes over Australia. These two areas are the primary and secondary targets, respectively, of the ADS-B payload, as discussed further in Section 2.4. A diagram of the nominal attitude profile of the spacecraft is shown in Figure 1.6, with the spacecraft throughout its orbit represented in orange and the ADS-B antenna boresight defined by the arrow. Magnetic field lines are shown in gray.

1.4.2 Deployable Drag Sails

CanX-7 was launched in September 2016 on a Indian PSLV rocket into a 670 km sun-synchronous orbit. After a seven month campaign of recording ADS-B messages, it was planned to deploy four triangular
drag sails into a square pattern in order to de-orbit the spacecraft. The 670 km orbit means that after the
drag sails are deployed, de-orbiting is estimated to take approximately 2.5 years [19]. With a lower orbit,
the satellite would deorbit more quickly, but 670 km is a relatively high altitude for a small satellite. As
such, the deorbiting phase will last longer, and the effect of the drag sails’ on the spacecraft orbit can
be measured more precisely. Two images of CanX-7 are shown in Figure 1.7, with the drag sails stowed
and deployed.

Figure 1.7: The CanX-7 spacecraft with drag sails stowed (left) and deployed (right)

1.4.3 ADS-B Receiver Secondary Payload

The CanX-7 ADS-B receiver payload was developed in partnership with the Royal Military College of
Canada (RMCC). Initial prototyping of the receiver on high-altitude balloons [25] was done by RMCC
using similar hardware to that which was flown on CanX-7, while the spacecraft payload was built and
tested at SFL. The flight payload consists of five major sub-components: an L-Band antenna, low-noise
preamplifier, ADS-B receiver, payload computer, and aluminum enclosure. The electronic components of
the ADS-B receiver are located within the enclosure in order to reduce electromagnetic interference from
the other spacecraft components, and the antenna is mounted to the exterior surface of the enclosure.
An exploded view of the ADS-B receiver payload is shown below in Figure 1.8.

The receiver component of the payload is a compact, single-board, commercial-off-the-shelf ADS-
B receiver. It contains an RF front end, logarithmic power detector, and an integrated FPGA. A
proprietary FPGA image handles decoding of the ADS-B messages and interfacing with the payload
computer. Upon demodulation and a successful CRC check, each message is tagged with a precise time
of arrival and signal strength measurement, and then stored in memory by the payload computer. Once a
recording is complete, the file containing the ADS-B data is then transferred to the spacecraft computer
for storage and downlink.

1.5 Thesis Scope and Motivation

The goal of this thesis is to present the work performed to take the CanX-7 ADS-B receiver payload
from the initial prototype developed by RMCC, to a flight-ready and operational payload. To do this,
there were several challenges that needed to be addressed.
First, sufficient performance and suitability of the prototype receiver for a space mission was verified. The payload needed to be sensitive enough to work at the long distances that would separate the spacecraft from any aircraft. This was done through simulation of the link between aircraft and the receiver, including a ground coverage analysis.

After the hardware chosen for the prototype payload was chosen, an antenna was designed in order to receive the ADS-B signals. A patch antenna was chosen based on both system-level and subsystem-level constraints, and was designed in tandem with another SFL student. The antenna was designed such that it would match the ADS-B frequency, provide sufficient gain to close a link with transmitting aircraft even in non-optimal pointing orientations (up to 45 degrees off-boresight), and respect size and mass limits placed upon it by the spacecraft bus and launch requirements. In addition, the antenna materials were also chosen to provide consistent performance across a wide range of temperatures.

Once the electronics were chosen and the antenna had been designed, they both went through a rigorous set of tests at the unit level designed to qualify them for operation in the space environment. Electronics underwent a full suite of thermal acceptance and thermal sensitivity tests, and the antenna gain pattern and other performance parameters were verified through pattern testing. Once the electronics and antenna had been individually verified, the payload had to be integrated into the spacecraft, and a system level electromagnetic compatibility test was performed in order to ensure that none of the spacecraft’s other systems interfered with the operation of the ADS-B payload or significantly reduced its sensitivity.

Software was also developed for the purposes of both controlling and operating the payload once it was integrated into the spacecraft, and for storing and managing its recorded data. The payload control software had to be compatible with the payload hardware while also being reliable and flexible.

The remainder of this thesis will be broken up into six parts, each focusing on a different aspect of
the work completed for this project. The parts are listed below.

- CanX-7 ADS-B Payload Background and Hardware Selection
- Spacecraft Communication Link Analysis Background
- Payload Ground Coverage Simulation
- Antenna Design and Testing
- Control and Ground Software
- Electronics Acceptance Testing
Chapter 2

Air Traffic Control Systems and the CanX-7 ADS-B Payload

In order to understand the context for the rest of this thesis, the following chapter will provide an introduction to the Automatic Dependant Surveillance-Broadcast (ADS-B) system for air traffic control, as well as an introduction to the ADS-B payload hardware.

The ADS-B standard grew out of the Federal Aviation Authority Surveillance Vision Plan’s recommendation to transition to a ground and satellite based air traffic control system in the mid 1990’s [26]. With better aircraft position determination accuracy, simpler infrastructure requirements, and the ability to provide situational awareness to aircraft in an ad hoc manner without interaction from a ground controller, ADS-B offers several improvements over the current air traffic control system. The first part of this chapter is dedicated to explaining the three primary Air Traffic Control (ATC) methods in use today, with particular emphasis on ADS-B. The second part of this chapter focuses on the CanX-7 ADS-B payload hardware, including a detailed description of the components that make up the physical payload.

2.1 Traditional Air Traffic Surveillance Techniques

There are several methods of tracking aircraft that are currently used by air traffic control organizations, with ADS-B being the newest and most advanced. Primary and Secondary Surveillance Radar (PSR and SSR, respectively) are also both widely used throughout most of the world for the tracking of aircraft in real time.

2.1.1 Primary Surveillance Radar

Primary surveillance radar (PSR) is an important part of aircraft traffic control and management in the areas surrounding medium to large sized airports [38]. Unlike Secondary Surveillance Radar and ADS-B, primary radar is a non-cooperative tracking method, meaning that it doesn’t rely on an aircraft having an operational transponder or a valid set of response data to determine the aircraft’s position. PSR is a monostatic, bi-directional system that works by transmitting a narrow-beam radar pulse from a directional antenna that is designed to bounce off the skin of an aircraft and return to the transmitting
antenna. When a return pulse is received, the radar knows that an aircraft has been detected and in what direction it is located. By measuring the time between when the radar pulse is sent and when the (typically much weaker) return signal is received, the distance to the aircraft can be calculated. With the direction and distance from the radar station, the aircraft can then be located. If the receiver is also equipped to measure the Doppler shift of the reflected radar pulse, the aircraft’s velocity relative to the radar station can also be determined. An illustration of this technique is shown in Figure 2.1.

![Figure 2.1: Illustration of the PSR technique](image)

PSR is highly useful because of its non-cooperative nature. Any aircraft (or other object in the air) can be tracked, irrespective of whether or not it is properly equipped with a correctly configured and working transponder. Primary radar, however, does have some big disadvantages. Even though it is possible, PSRs are not typically designed to be able to determine the altitude of an aircraft. PSR also does not give an air traffic controller any information about the aircraft which has been detected. There is no way for the controller to tell what type of aircraft is being detected, its condition, flight number, or even whether it is an aircraft at all without a secondary form of communication with the aircraft or its pilot. Until the advent of secondary radar, correlation of aircraft call signs to radar tracks was a manual process heavily dependent on air traffic controllers and aircraft pilots [43]. Inherent in the system is also the need for a large and expensive ground infrastructure. Ground-based antennas need to be highly directional in order to give an accurate position reading for each detected aircraft, and are typically large and expensive, with extremely high power output [43].

Depending on the required range, an extremely high output RF power level is normally needed for a PSR to function properly. Equation 2.1, also known as the radar equation, governs the power of the received radar signal after it has scattered off a target aircraft [15]. It can be seen that the most sensitive term in the equation is the range from the radar station to the aircraft, $R$, which is raised to the fourth power. The radar equation defines the expected received power $P_r$ of a radar pulse transmitted at a power $P_t$ of wavelength $\lambda$ from an antenna with a gain of $G$, which has reflected off of an object with radar cross section $\sigma$. The system also experiences fading based on environmental factors such as rain, and can also be disrupted by land features too close to the radar beam. For the PSR to be able to detect an object, the received signal power must be high enough such that the system can easily distinguish a reflected signal in the presence of noise.
The directionality of the antennas used for primary radar is important for determining an aircraft’s position, but it is also important for increasing the transmitter and receiver gain $G$. The massive losses due to spherical spreading of both the transmitted and reflected signals must be made up for with extremely high power output, very high gain antennas, and extremely sensitive receiver equipment. Due to the costs and complexity associated with these systems of large antennas with powerful amplifiers and highly sensitive receivers, they are typically only installed in the areas around large airports where traffic density and the risk of collisions is high. They do not provide a good option for expanding ATC coverage across large areas of the Earth that currently lack aircraft tracking coverage.

### 2.1.2 Secondary Surveillance Radar

The most widely used air traffic control method at airports is currently the Traffic Collision Avoidance System, or TCAS. A form of SSR, TCAS is a form of cooperative surveillance, and as such requires compatible equipment to be installed and functioning on an aircraft in order for the system to work. Figure 2.2 demonstrates how SSR works, with an interrogation message transmitted by a directional antenna, and a reply message transmitted by an omni-directional antenna on board the aircraft.

![Illustration of SSR technique](image)

**Figure 2.2:** Illustration of the SSR technique

A TCAS transmitter works in a similar manner to primary radar, but it uses a two part approach to achieve similar results. Rather than relying on signals reflected from the skin of an aircraft to establish its presence, the system relies on aircraft receiving interrogation messages and responding to them. A typical TCAS antenna has a narrow beamwidth similar to that of a PSR antenna, and constantly transmits an interrogation message to nearby aircraft, asking for replies. The beam of the antenna is shaped such that the interrogation message is picked up by aircraft within a narrow azimuthal range and a large range of altitudes. When the message is received by an aircraft’s transponder, it transmits a reply containing an identification code for the aircraft and its current altitude. Based on a time of flight measurement between sending the interrogation and receiving the reply (distance), altitude information contained in the reply (elevation), and the antenna’s current pointing direction (azimuth), aircraft can be pinpointed by the ground station with relative accuracy in 3D space. The aircraft that are identified via this means are also identified by a code unique to each aircraft in the vicinity [43], so it is far easier...
for the air traffic controllers to determine which aircraft is which.

This method offers three advantages over primary radar. The first is that by not relying on the measurement of a reflected signal and instead having a reply transmitter on board each aircraft, the transmitter power and receiver sensitivity that are required of the ground station can be lower than that of a primary radar while achieving the same detection range. Second, the altitude of the aircraft can be found within the reply message of the TCAS system, resulting in a more precise position determination. Third, the TCAS system includes within its response message a way to identify each aircraft more precisely and with less work from air traffic controllers. This speeds up the identification of particular aircraft immensely, and reduces the amount of work that must be done by controllers. This type of system is very useful in high-traffic areas, where controllers would otherwise have a difficult time keeping track of all the aircraft.

2.1.3 Primary and Secondary Surveillance Radar Coverage

Due to the high costs associated with operating a station, it is not always practical or economical to install enough PSR or even SSR stations to provide coverage over sparsely populated areas. Figure 2.3 shows the extent of PSR and SSR coverage over Canada. It can be seen that while most populated areas are covered by at least one of the two types of surveillance radar, many areas in the far north and over Hudson Bay lack coverage by either method. As many flights cross through these areas, enhanced coverage in them is an important part of the next generation of aircraft monitoring.

![Figure 2.3: Primary (left) and Secondary (right) Radar Coverage In Canada [38]](image)

2.2 Automatic Dependant Surveillance-Broadcast

Automatic Dependant Surveillance-Broadcast (ADS-B) is designed to be a key part of the next generation ATC system. First tested in Australia in 2003 [43], it is a cooperative surveillance technique that addresses many of the issues with PSR and SSR. While primary and secondary radar (TCAS) require all position determination to be performed by a ground station, ADS-B requires an aircraft to use on-board position source independent of any interaction with the ground for determination. While the particular method of position determination is not specified, it must meet certain accuracy standards
Chapter 2. Air Traffic Control Systems and the CanX-7 ADS-B Payload

based on the airspace it operates in, with most aircraft using a GNSS receiver of some sort. At a semi-
random interval, position, velocity, and other information is automatically transmitted to other nearby
aircraft and ground stations in the vicinity. This system determines aircraft position more accurately
than either PSR or SSR, and is more flexible in that it can be used by multiple aircraft in an ad hoc
manner, without any ground infrastructure.

Because it is not dependent on a ground station to operate, ADS-B is unique among ATC techniques.
An ADS-B transmitter on an aircraft will transmit many different types of message, but the most
common (and those the CanX-7 is most interested in recording) are aircraft position, velocity, and flight
information messages. Position messages are transmitted by the aircraft twice every second, alternating
between top and bottom-mounted ADS-B antennas [44]. These messages contain the transmitting
aircraft’s International Civil Aviation Authority (ICAO) identification address, the aircraft’s altitude
precise to either 25 ft (below 50,175 ft) or 100 ft (50,175 - 203,700 ft), and a portion of the aircraft’s
latitude and longitude in a format called Compact Position Reporting (CPR) [42]. Velocity messages
contain the same identification information as position messages, and are also transmitted twice per
second, alternating between the top and bottom-mounted antennas. Flight identification messages are
broadcast even less frequently, once every five seconds [42]. These are not the full extent of messages
transmitted by an ADS-B system, but are some of the most important for maintaining adequate aircraft
spacing.

The biggest advantage of this system is that it removes the need for large, directional antennas and
transmitting hardware. A simple, omni-directional antenna can be paired with a relatively inexpen-
sive receiver and replicate all the functions of a secondary surveillance radar station, but with even
greater precision. The disadvantage of the system is that it can easily be spoofed by a transmitter
broadcasting false information. Since there is no encryption or message authentication on the channel,
a non-cooperative target could broadcast a false location to receivers. The threat of this happening is
mitigated through the use of multilateration, a technique which uses the time difference of arrival of an
ADS-B message at three different receivers spaced widely apart to calculate the location of an aircraft.
This calculated triangulation of the aircraft can then be verified against the position that was broadcast
to ensure that false aircraft are identified and filtered out of the data stream used by the air traffic
controllers.

2.2.1 Aircraft Transmitter Hardware and Regulations

As a part of the ADS-B standard, aircraft equipment must comply with a set of standards designed
to ensure that the ADS-B system functions properly. There are guidelines for both transmitting and
receiving hardware on aircraft, but only the transmitting hardware will be discussed here. All ADS-B
systems must be connected to a GNSS system providing a position measurement and have a transmission
latency no more than 2 seconds between position determination and broadcast, with only 0.6 seconds
of the latency not being compensated for [24]. In other words, an aircraft must be able to broadcast
a location determined no more than 2 seconds in the past. If, however, the latency is greater than 0.6
seconds, the movement of the aircraft since position determination must be compensated for such that
the position broadcast is of a similar accuracy to one with a latency of no more than 0.6 seconds. The
position determination must be accurate to a level determined by the class of airspace the aircraft is
operating in. Large commercial aircraft, those CanX-7 is most interested in tracking, spend most of
their time in what would be considered FAA class A airspace, from 18,000 ft up to Flight Level 600
(FL600), approximately 60,000 ft above sea level. This means they must maintain a horizontal position determination error of no more than 92.6 meters (0.05 nautical miles) at all times [28].

There are also requirements placed on the the transmitters and transmitting antennas. These standards are particularly important as they are used as the basis of much of the analysis performed in Chapter 4 of this thesis. ADS-B broadcast power at the antenna input must be between 125 and 500 Watts [28], and each equipped aircraft must carry a bottom-mounted antenna or both top- and bottom-mounted antennas. While most large aircraft will carry both, aircraft are only required by the FAA to carry one on the underside of the aircraft to be compliant with regulations [24].

2.2.2 ADS-B Physical Layer

The remainder of this section describes the ADS-B signal using the Open Systems Interconnection (OSI) model as a framework. Only two layers of the seven layers in the OSI model are relevant to ADS-B, as it is a point-to-point broadcast protocol in which data is not routed between nodes. The two layers that the ADS-B standard defines are the physical layer (layer 1) and data link layer (layer 2).

A link’s physical layer is defined by the electrical specifications that make up the communication standard. For ADS-B, this includes the center frequency, modulation type, and modulation rate. ADS-B is transmitted at a center frequency of 1090 MHz, with a data rate of 1 Mbps. Data is modulated onto the carrier wave using an amplitude modulation scheme called Binary Pulse-Position Modulation (Binary PPM), described further in Section 2.2.2 and Figure 2.4. For all intents and purposes, the physical layers of mode-S ADS-B messages and mode-S TCAS replies are the same [45]. This means that the two formats can decoded by the same hardware, making ADS-B’s integration into the existing ATC infrastructure simpler.

**ADS-B Modulation**

The modulation scheme of the ADS-B physical layer is one of the key aspects that enables the system to work efficiently. Data is modulated in the baseband signal via Binary PPM. PPM is a form of amplitude modulation whereby data is encoded by sending pulses at specific times to represent binary 0’s and 1’s. Each data symbol is specified by the particular user to be a certain length, in the case of ADS-B, one microsecond. Each symbol is then broken up into \(2^M\) time slots, where M is the modulation order. For binary PPM, the modulation order \(M = 1\), so each symbol is broken up into two time slots. When a binary 1 is transmitted, a pulse is sent in the symbol’s first time slot, while a binary 0 is represented by a pulse in the symbol’s second slot.

PPM is not often used in long distance radio communications, as it is highly susceptible to multipath interference, and as a non-coherent modulation scheme, requires a higher signal-to-noise ratio to decode than a coherent one. In many terrestrial cases, the disadvantages associated with multipath interference make PPM a poor choice. However, because many aircraft have to share the ADS-B channel and because the data messages are so short and broadcast intermittently, there wouldn’t be sufficient time before the end of the message for a receiver to establish a phase lock and decode the data if a coherent modulation scheme were used.

Even though PPM doesn’t require a receiver to establish a phase lock to decode data, the receiver still needs a point of reference for each message to be compared against so the data can be correctly interpreted. In ADS-B, this is done through the use of a message preamble. Figure 2.4 shows an example
Chapter 2. Air Traffic Control Systems and the CanX-7 ADS-B Payload

Figure 2.4: ADS-B baseband binary pulse-position modulation [34]

of a modulated mode-S extended squitter data packet at baseband. The packet begins with an 8 $\mu$s preamble and contains either 56 or 112 data bits, for a maximum of length of 120 $\mu$s. The preamble is made up of four pulses, one in each of the slots beginning at 0, 1, 3.5, and 4.5 $\mu$s after the start of the message. While it looks similar to data at baseband, the preamble does not actually represent any data using the PPM scheme. Instead, it is used as a constant reference for the receiver to synchronize the demodulation of data bits to.

2.2.3 ADS-B Data Link Layer

The data link layer defines the protocols used for node-to-node data transfer. In this case, the two nodes are a transmitting aircraft and a receiver, of which there can be many. After the preamble, data is transmitted in one of two types of messages, either an acquisition squitter (short frame) or extended squitter (long frame) [23]. Both types are effectively identical except for an extra 56-bit data field contained in extended squitter messages. Figure 2.5 shows how the ADS-B message is broken down on a high level, with different shading colors representing data fields, red lines representing byte divisions, and black lines representing bit divisions within the bytes. A long frame consists of 112 bits (14 bytes), while a short frame is missing the data field in bytes 5-11 and is only 56 bits (7 bytes) long.

Figure 2.5: Breakdown of ADS-B frame data

The message begins with the downlink format and aircraft flight status. The downlink format is 5 bits long, and indicates the type of message being transmitted. ADS-B uses downlink format 17 for extended squitters and downlink format 11 for acquisition squitters. The next three bits indicate whether the aircraft is in the air, on the ground, and whether there is an alert on board the aircraft. One of the bits also is a special indicator bit, which can be turned on (set to 1) by the pilot at an air traffic controller’s
request as part of an aircraft identity confirmation. The next three bytes in the data message identify an aircraft with its 24-bit ICAO address, a unique identifier given to each aircraft that is nominally never changed, and at the end of the message a 24-bit parity check code is included for error detection.

Extended squitter messages each contain a 56 byte data message between the ICAO address and the parity information. Within the data block, the first five bits denote the type of message, with the remaining bits used for the actual data being transmitted. There are many types of data messages containing different types of aircraft and flight data. For more information on the types of data that are broadcast, see Appendix 1 of [28].

2.3 CanX-7 Payload Hardware Configuration

During the preliminary design phase of the CanX-7 mission, an ADS-B receiver was chosen to be CanX-7’s secondary payload to demonstrate the tracking of aircraft from space. As SFL has flown numerous spacecraft with Automated Identification of Ships (AIS) receivers, it was a natural progression to move into other forms of radio tracking. The payload was developed in partnership with the Royal Military College of Canada (RMCC), with much of the initial prototyping and testing performed by RMCC and final flight development and qualification testing performed by SFL.

2.3.1 Royal Military College of Canada Prototype

In preparation for flight, the prototype receiver developed by RMCC was delivered to SFL in January of 2015. The prototype consisted of four elements:

- Linux-based microcontroller
- Commercial off-the-shelf ADS-B receiver
- Low-noise RF preamplifier
- Microstrip patch antenna

The antenna was in the early stages of development, but the microcontroller, ADS-B receiver, and preamplifier had all been extensively tested in high-altitude balloon experiments by RMCC [25]. This payload was used by SFL for the initial assembly of the spacecraft outside of the clean room, known as dirty-sat integration, or just dirty-sat. The electronics are shown in their enclosure in Figure 2.6.

2.3.2 Improvements to Prototype Payload

With the prototype payload in hand, work was begun on building a flight-ready version for CanX-7. Updated versions of the microcontroller and receiver hardware were purchased from the manufacturer, and a slightly different model of the preamplifier with a stabilizing network to improve the stability factor but similar gain and noise figure performance was purchased to replace the original.

The new receiver contained some minor upgrades to the design, most notably it removed a set of physical switches from the top of the board used to configure the FPGA and software parameters, and replaced them with the ability to configure the same parameters in software. As hardware switches can’t be reset once in space, this new receiver offered additional flexibility compared to the prototype. The
microcontroller was also upgraded to a newer revision of the board with a lower power consumption, a faster processor, more memory, built-in storage, and additional I/O ports. Before any testing the payload or developing software was done, some modification of the microcontroller and receiver was necessary in order to better fit the payload within the mass and power constraints of the spacecraft.

Figure 2.7 highlights the modifications made to the receiver and microcontroller payload computer. On the receiver, the SubMiniature version A (SMA) coaxial input connector was removed and replaced with a U.FL connector due to mechanical restrictions within the payload enclosure. A temperature sensor was added and soldered to an unused header pin that was connected to the payload computer’s analog to digital converter (ADC), and its Global Positioning System (GPS) receiver chip was removed from the board in order to reduce power consumption. On the payload computer, most of the connectors were removed from the board to allow it to fit into the enclosure, along with two unneeded tactile switches. Power provided by the spacecraft power system was delivered through a power/ground wire twisted pair soldered directly to the connections of the barrel connector, while serial lines and a twisted pair of power/ground wires connected to the preamplifier were soldered directly to the appropriate header pins. It should be noted that with the exception of the addition of the temperature sensor and the removal of the GPS receiver chip, all of these modifications were present on the prototype payload avionics as well.

Antenna

Other than the metal payload enclosure, the ADS-B receiver antenna is the payload’s only custom-designed piece of hardware. When the type of antenna to use was chosen, several factors had to be taken into account before a specific design was chosen. The requirements for gain pattern, polarization, and bandwidth, as well as system-level mechanical, thermal, and attitude control constraints were all taken into account when making a decision. The final design chosen for the payload was a circularly-polarized probe fed patch antenna, similar to the prototype, except that during the design process high gain and beamwidth were prioritized over narrow frequency bandwidth. More details on design decisions made regarding the final antenna can be found in Chapter 5.
Chapter 2. Air Traffic Control Systems and the CanX-7 ADS-B Payload

2.4 CanX-7 Payload Concept of Operations

The primary mission of the CanX-7 ADS-B receiver is to demonstrate reception of ADS-B messages from aircraft crossing the North Atlantic Ocean with a commercial-grade ADS-B receiver aboard a nanosatellite. The original concept of operations for the payload was to operate it for a total of two minutes per orbit, collecting data over the North Atlantic Ocean if the orbit crossed near it, or over another location if it did not. Two minutes of operations per orbit results in a duty cycle of less than 5%. While maintaining this would meet mission requirements, the goal of the modifications to the hardware was to obtain better sensitivity while simultaneously increasing the amount of time that the payload could operate each orbit before running into power or thermal limitations. The payload computer was updated to a newer, more efficient model, while the GPS receiver was removed to decrease current draw and heat generation. With the prospect of running the payload longer than initially planned, the addition of a temperature sensor enabled the use of a safety cut-off if the board began running too hot during extended operations.

Based on these hardware modifications, modifications to the flight software discussed in Section 6.6.1, and better than expected power generation while on orbit, payload operations were able to be extended from 2 minute recordings, once per orbit, to 18 minute recordings once per orbit, dramatically increasing the amount of data that was able to be collected during the mission.
Figure 2.8: CanX-7’s ADS-B payload antenna
Chapter 3

Link Analysis Background

The link budget is one of the most important tools in communication system analysis. It is used to add up and keep track of all of the gains and losses a signal experiences between a transmitter and receiver. The link budget is often the first step in analyzing a communication system, and is used to determine if the link will close based on the system parameters. Once the expected signal power and noise power at the receiver are calculated, a signal-to-noise ratio (SNR) dependent on the system modulation (usually required bit energy to noise density ratio or signal-to-noise ratio) is subtracted from the expected SNR of the input signal at the receiver in order to ensure a specified bit-error-rate can be met by the proposed system. The difference between the value of the calculated figure of merit and the figure of merit required for the link to close is called the link margin. When designing a communications system, the ability to meet a specified link margin is typically one of the key requirements, with an adequate margin ensuring that a link will function as intended even if real-world conditions turn out to be worse than those assumed during the design phase.

The components of a link budget will be broken up into three sections in order to organize this chapter. Splitting the components of the link budget up in this manner both makes sense from a topology point-of-view and will reduce the amount of redundant calculation required for the more complex analysis performed in Chapter 4. Link budget terms are grouped into those dependent on the transmitter, those dependent on the environment and the geometry of the link, and those dependent on the receiver. All individual terms in Sections 3.1 through 3.3 will be calculated as linear values rather than logarithmically, unless otherwise stated. Logarithmic values are used in Section 3.4 for simplicity when combining terms. An example link budget for the CanX-7 ADS-B payload is provided in Figure 3.1 as a reference.

3.1 Transmitter

All radio links have their point of origin at the transmitter. In a link budget, the terms considered as part of the transmitter segment are transmitter power $P_{TX}$, antenna gain $G_{TX}$, and signal polarization (losses from which are expressed as $L_{PTX}$). The goal of the transmitter section of a link budget is to determine the Effective Isotropic Radiated Power ($EIRP$) of the transmitter. This value, typically expressed logarithmically in decibel-milliwatts or decibel-Watts (dBm or dBW), is found by combining the effective power of a signal radiated by a transmitter with any gains or losses from amplifiers, antennas, filters, and other components in the transmit chain. It represents all of these components in the transmit
chain by treating them as though they are a transmitter with an output power equal to the EIRP value connected losslessly to an isotropic antenna.

### 3.1.1 Transmitter Power

All radios work by converting direct current (DC) into a sinusoidally oscillating alternating current (AC), which is converted into radiated radio frequency (RF) energy in an antenna. The transmitter power is the amount of electrical energy which is sent to the antenna, typically measured linearly in mW or logarithmically in dBm. It is important to distinguish this power value from the DC power required for the transmitter to function, as DC to RF power conversion is not 100% efficient. This efficiency is important to take into consideration when choosing or designing a transmitter, as one with an efficiency of 30% will have an additional 3 dB of radiated power (and therefore link margin) compared to one with
15% efficiency, while drawing the same amount of DC power.

### 3.1.2 Bit Rate and Energy Density

A transmitter encodes information onto a carrier signal by modulating it in such a way that analog or digital information can be reclaimed by a receiver. For instances where lossless data transmission isn’t necessary and simplicity in the transmit-receive system is required, such as some radio and television broadcast sources, analog modulation may be used, but most modern satellite communication systems are used for lossless transmission of data. Digital modulation schemes are designed to encode information bits onto a carrier signal, which can then be reconstructed even in the presence of a significant amount of noise through error detection. The rate at which the modulation scheme encodes information bits onto the carrier signal is referred to as the bit rate, and is an important factor in link analysis.

When an unmodulated sinusoidal carrier signal is transmitted, all of the power in the signal is focused at that frequency. Without any modulation, a carrier signal is theoretically detectable at any power level, given a long enough integration time in the absence of non-Gaussian noise, as a perfect carrier would have a bandwidth of 0 Hz. When a carrier signal is modulated, however, the power is spread across the frequency spectrum rather than being concentrated at a point [36]. In order to be detectable, the signal power must be above the power of any noise also detected by the receiver over the frequency range it is being modulated across. As a general rule of thumb, the faster the carrier signal is modulated, the more the power is spread across the frequency spectrum, and the closer it is to the noise level. Figure 3.2 shows two signals transmitted at the same level, where one is modulated and one is unmodulated. It can be seen that though they both have the same channel power, the modulated signal is spread out across a greater portion of the frequency spectrum.

![Image](image.png)

Figure 3.2: An unmodulated carrier signal (left) and one modulated at 1 MHz using PPM (right)

The relationship between data rate and occupied bandwidth is dependent on the type of modulation used. Typically, coherent modulation schemes such as phase shift keying (PSK) will occupy less bandwidth for a given data rate than incoherent modulation schemes, such as frequency shift keying (FSK), but are more complicated to implement in receiver hardware [27].

### 3.1.3 Transmitting Antenna

The conversion of electrical power into RF power happens in an antenna, where oscillating electrical current induces an electromagnetic field around a conductor. While oscillating current in any conductor
will generate electromagnetic waves, an antenna is a structure specifically designed to efficiently radiate in one or many directions over a designed frequency range.

**Antenna Gain**

An antenna will concentrate its energy in different directions based on it’s geometry and the materials that it’s made from; the way that the radiated energy is concentrated is called its directivity. The pattern that defines the directionality of an antenna is often referred to as its radiation pattern or gain pattern. This pattern is typically determined by measuring the amount of radiated power coming from the antenna in a given direction and referencing that radiated power against either the peak gain of the antenna or another constant reference. When quantifying an antenna’s radiated power, it is common to measure it logarithmically and reference it to an isotropic antenna, an idealized lossless antenna which collects RF power equally in all directions. The value defining the combination of the antenna’s directionality and efficiency in a specific direction is called gain, and can be measured in dBi, or decibels relative to an isotropic antenna. Referencing a gain pattern to an isotropic antenna rather than the antenna’s peak gain is useful because it allows for better comparison between multiple different antennas.

**Isotropic Antenna**

While a perfect lossless isotropic radiator does not exist in practice, it provides an important reference for several equations and measurements throughout the rest of this thesis. As stated previously, an isotropic antenna is a lossless antenna that collects (and therefore also radiates) energy equally in all directions at a specified frequency. This means that the antenna’s gain is unity across $4\pi$ steradians around the antenna, and the amount of energy collected from an electromagnetic field with uniform flux is the same regardless of the incoming direction of the waves generating the field. Having a unity gain, however, says nothing of the amount of energy that is actually collected by the antenna, only that it is the same in all directions.

A convenient way to define the amount of energy collected by an antenna independent of the magnitude of the flux of the electromagnetic field is by using the effective aperture of the antenna. The effective aperture (or collecting area) of an isotropic antenna for a specific wavelength $\lambda$ is defined by Equation 3.1 [37].

$$\langle A_e \rangle = \frac{\lambda^2}{4\pi}$$

This has several implications, the most important of which is that an isotropic antenna of a higher frequency (and thus a lower wavelength) will have a smaller effective aperture. Since the effective aperture is smaller, less energy overall can be absorbed by the antenna. This effective aperture discrepancy is important when calculating free-space loss in a link budget, further explained in Section 3.2.1.

**Antenna Polarization**

When RF energy is radiated from an antenna, it is done so in the form of a sinusoidal electromagnetic wave where the wave’s electric and magnetic fields oscillate perpendicular to each other and the direction of travel. Figure 3.3 shows an electromagnetic wave with wavelength $\lambda$ travelling in direction $z$, with the electric field oscillating in axis $E$ and the magnetic field oscillating in axis $B$. A wave’s polarization defines the manner in which its electric field oscillates with respect to the direction of travel. In order
to describe the polarization of an electromagnetic wave in a general sense, a wave’s polarization ellipse can be used.

Figure 3.3: Diagram of an electromagnetic wave [47]

Fresnel’s Theory [20] describes the propagation of a radiated electromagnetic wave, and specifically its electric field. When a radiated wave travels, the amplitude and direction of its electric field are expressed by a vector. Looking at the wave head-on from the direction towards which it is travelling, it can be seen that the magnitude of this vector oscillates sinusoidally as the wave moves through space. This oscillation of the electric field is not limited to one dimension, either. This electric field vector can be expressed as the combination of two sinusoids oscillating in the horizontal and vertical planes, $E_x(z,t)$ and $E_y(z,t)$.

\[ E_x(z,t) = E_{0x} \cos (\tau + \delta_x) \]  
\[ E_y(z,t) = E_{0y} \cos (\tau + \delta_y) \]

In Equations 3.2 and 3.3, $\tau$ represents the product of frequency and time, $E_{0x}$ and $E_{0y}$ represent the magnitudes of the electric field, and $\delta_x$ and $\delta_y$ represent the phase offsets of the respective sinusoids. By dividing by their respective magnitudes and using the angle sum identity, these equations can begin to be expanded and combined, with the aim of removing the frequency and time components.

\[ \frac{E_x}{E_{0x}} = \cos \tau \cos \delta_x - \sin \tau \sin \delta_x \]  
\[ \frac{E_y}{E_{0y}} = \cos \tau \cos \delta_y - \sin \tau \sin \delta_y \]

Rearranging equation 3.4 to solve for $\cos \tau$ and equation 3.5 to solve for $\sin \tau$ results in equations 3.6 and 3.7.

\[ \cos \tau = \frac{E_x}{E_{0x}} + \sin \tau \sin \delta_x \cos \delta_x \]  
\[ \sin \tau = \frac{\cos \tau \cos \delta_x - E_y}{E_{0y}} \]

By substituting Equations 3.6 and 3.7 into Equations 3.5 and 3.4, respectively, each equation is now be expressed only in terms of $\sin \tau$ and $\cos \tau$. Rearranging and using the angle sum identity once again
gives equations 3.8 and 3.9.

\[
\frac{E_x}{E_{0x}} \sin \delta_y - \frac{E_y}{E_{0y}} \sin \delta_x = \cos \tau \sin (\delta_y - \delta_x) \tag{3.8}
\]

\[
\frac{E_x}{E_{0x}} \cos \delta_y - \frac{E_y}{E_{0y}} \cos \delta_x = \sin \tau \sin (\delta_y - \delta_x) \tag{3.9}
\]

By squaring both sides of equations 3.8 and 3.9, adding them together, and combining like terms, the horizontal and vertical sinusoids are now combined into a single equation. By applying the Pythagorean and angle difference identities, the resulting terms can be arranged into Equation 3.10, eliminating all terms containing the time \( \tau \), meaning the result is time-independent. The result is also now in the form of the equation of an ellipse, where \( \delta = \delta_y - \delta_x \) is a constant that represents the phase offset between the vertical and horizontal sinusoids. This equation for an ellipse which can succinctly describe the polarization of the wave.

\[
\frac{E_x^2}{E_{0x}^2} + \frac{E_y^2}{E_{0y}^2} - 2 \frac{E_x}{E_{0x}} \frac{E_y}{E_{0y}} \cos \delta = \sin^2 \delta \tag{3.10}
\]

Equation 3.10 shows the general form of the polarization ellipse, normalized by the magnitudes of the original sinusoids \( E_{0x} \) and \( E_{0y} \). This ellipse follows the general form for the equation of an ellipse, \( Ax^2 + Bxy + Cy^2 + Dx + Ey + F = 0 \). This ellipse can be used to describe the polarization of any wave emitted by an antenna, and thus the polarization of the antenna itself. A plot of an example polarization ellipse can be seen in Figure 3.4, where the angle \( \beta \) is a measure of the ellipticity caused by the difference in the magnitudes of \( E_{0x} \) and \( E_{0y} \) and the rotation angle \( \chi \) is caused by the cross term in Equation 3.10 [20].

![Figure 3.4: An example polarization ellipse](image)
Degenerate Polarization States

Elliptical polarization is the general form of polarization, but there are several special cases which are commonly used by spacecraft communication systems. These forms are called degenerate polarization states [20].

There are three types of degenerate polarization: linear, diagonal, and circular. The simplest type of polarization is linear polarization, which can be further divided into horizontal or vertical polarizations. Horizontal and vertical polarizations exist when one of either $E_{0y}$ or $E_{0x}$ from Equations 3.5 or 3.4 are equal to zero. This causes the polarization ellipse to appear as a line along the x-axis for horizontal polarization or y-axis for vertical polarization. Diagonal polarization is similar to linear polarization in that its ellipse also appears as a line, but it is rotated $+/- 45^\circ$. Mathematically, this happens when $E_{0x} = E_{0y}$ and $\delta = 0$ or $\pi$ [20].

Circular polarization exists when $E_{0x} = -E_{0y}$ and the vertical component of the oscillation is exactly $+/- 90^\circ$ out of phase with the horizontal component ($\delta = +/- 90^\circ$). Circular polarization can be either right-handed or left-handed, depending on which direction the polarization vector rotates as the wave propagates. If $\delta$ is equal to 90°, the wave is right-handed circularly polarized, and if $\delta$ is equal to -90°, it is left-handed circularly polarized.

Applications of Different Polarizations

The polarization chosen for a particular application is dependent on a number of different factors. Most spacecraft communications will use some form of circular polarization, as polarization losses between two ideal antennas of similar handedness will be zero regardless of their relative orientations. In reality, no two antennas will be perfectly circularly polarized in all orientations, so there is some loss due to polarization mismatch, but it will be small if the handedness of the polarization of the two antennas is the same. This is very useful for a communications system where one of the antennas (the one on the spacecraft) will not necessarily always be in a known or controlled orientation.

Linear polarization is used more often in earth-based communication systems, such as AIS and ADS-B, where the typical orientation of the antennas is predictable and consistent, and where relatively simple antenna designs produce omni-directional patterns with good performance focused at the horizon. Both ships and commercial aircraft tend to spend a majority of their time in level seas or flight, respectively, meaning the benefits of circular polarization are outweighed by the omni-directional nature of a linearly polarized omni-directional antenna. More information about the losses associated with polarization mismatch can be found in Section 3.3.1.

Antenna Frequency and Bandwidth

The essential purpose of an antenna is to serve as a matching network between the transmitter and free space, allowing for effective and efficient power transfer between the two. An antenna will not operate equally well over all frequencies, however. Based on its design, each antenna has a range of frequencies for which it will radiate efficiently, known as the antenna’s bandwidth (also called impedance bandwidth). The frequency at the center of the bandwidth, typically the one where the antenna radiates most efficiently, is called the center frequency.

To transfer power between a transmitter and an antenna, they are connected via a transmission line, most often a coaxial cable in space applications, whose impedance is chosen based on application. Most
coaxial cables have an impedance of either 50 or 75 Ω, with 50 Ω being more common. Given that free-space has an impedance of 377 Ω, it may initially seem strange that coaxial cables would not be designed to match the impedance of free space. A 377 Ω coaxial cable, however, would be extremely lossy. Equation 3.11 defines the impedance $Z_0$ of a coaxial cable based on its outer conductor diameter $D$, inner conductor diameter $d$, and dielectric constant $\epsilon_R$. Equation 3.12 defines the loss of the cable based on the impedance $Z_0$, the frequency $f$, and the properties of the outer (1) and inner (2) conductors, $\mu_R$ and $\rho$. $\mu_0$ is the permeability of free-space.

$$Z_0 = \frac{138}{\sqrt{\epsilon_R}} \log_{10}(\frac{D}{d}) \quad (3.11)$$

$$L_{\text{cable}} = \frac{8.686}{2Z_0} \sqrt{\frac{\mu_0}{\pi}} \left( \sqrt{\frac{\mu_R \rho_1}{D}} + \sqrt{\frac{\mu_R \rho_2}{d}} \right) \quad (3.12)$$

The standard coaxial cable impedance of 50 Ω was chosen as a compromise between the maximizing the peak power handling ability and minimizing the loss of a coaxial cable using an air dielectric, common in high-power radio transmitters in the 1930’s [8]. With power from a transmitter entering over a 50 Ω coaxial cable, the center frequency of the antenna is the point at which it most efficiently matches the impedance of the cable to that of free space [5]. The bandwidth is then the range of frequencies over which the antenna will match the input power to free space and radiate it, bounded on either side by the a maximum reflected power limit. Depending on the type of antenna, impedance bandwidth can range widely. Horn and log-periodic antennas have very wide impedance bandwidths relative to their center frequencies, while the bandwidths of Yagi-Uda and patch antennas are relatively low.

**Antenna Types**

There are many types of antenna, each with different form factors and gain patterns, used in different applications. Shown in Table 3.1 is a selection of some of the most common types of antennas, with their polarization and directionality listed.

<table>
<thead>
<tr>
<th>Antenna Type</th>
<th>Type</th>
<th>Polarization</th>
<th>Directivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monopole</td>
<td>Wire</td>
<td>Linear</td>
<td>Omni-Directional</td>
</tr>
<tr>
<td>Dipole</td>
<td>Wire</td>
<td>Linear</td>
<td>Omni-Directional</td>
</tr>
<tr>
<td>Helical (normal mode)</td>
<td>Wire</td>
<td>Linear</td>
<td>Omni-Directional</td>
</tr>
<tr>
<td>Helical (axial mode)</td>
<td>Travelling Wave</td>
<td>Circular</td>
<td>Directional</td>
</tr>
<tr>
<td>Yagi-Uda</td>
<td>Travelling Wave</td>
<td>Either</td>
<td>Directional</td>
</tr>
<tr>
<td>Log-Periodic</td>
<td>Log-Periodic</td>
<td>Linear</td>
<td>Directional</td>
</tr>
<tr>
<td>Patch</td>
<td>Microstrip</td>
<td>Either</td>
<td>Hemispherical</td>
</tr>
<tr>
<td>Horn</td>
<td>Aperture</td>
<td>Either</td>
<td>Directional</td>
</tr>
<tr>
<td>Parabolic</td>
<td>Reflector</td>
<td>Either</td>
<td>Directional</td>
</tr>
</tbody>
</table>
3.2 Environment and Geometry Effects

After a signal is emitted from a transmitting antenna, it must travel through free space before reaching a receiver. Along the way, the signal power is attenuated via geometric spreading and atmospheric attenuation. In a link budget, the free space loss term typically provides the largest amount of attenuation to the signal. Depending on the frequency used, the atmosphere can attenuate signals severely, or allow them to pass through relatively unaffected.

3.2.1 Free-Space Path Loss

Any signal emitted from an antenna into free space will travel away from its source, spreading spherically as it travels. This loss of strength as the signal travels through free space is the single largest contributing loss to the link budget in most cases. It is important to note that free space loss is not really attenuation of the signal. If a signal is transmitted with 1 Watt RF power and the effects of atmospheric attenuation are assumed to be negligible, the total power flux across a sphere 1 meter in radius will be the same as a sphere 1 kilometer in radius: 1 Watt. The power flux density, however, will decrease proportionally with the increasing surface area of the sphere, and so will be much smaller in the second case. Even though the free-space path loss is not a true loss term, it is referred to as a loss by convention and so will be for the remainder of this thesis.

It would seem that the correct way to quantify free-space loss would be to calculate it based on the decrease in the power flux density, defined by the formula for the inverse surface area of a sphere, Equation 3.13. However, because a link budget often needs to incorporate the gain values of dissimilar transmitting and receiving antennas, it is desirable to quantify antenna gain based on decibels relative to an isotropic antenna (dBi). Isotropic antennas, however, have an effective aperture defining how much energy they can absorb for a given frequency. In order to reference the antenna gains in the other parts of the link budget to an isotropic antenna, Equation 3.13 must be multiplied by the effective aperture of an isotropic antenna of the correct frequency, Equation 3.1, to obtain the value of free space loss, Equation 3.14.

\[
\frac{1}{A_{\text{sphere}}} = \frac{1}{4\pi r^2} \quad (3.13)
\]

\[
L_{\text{fsl}} = \frac{1}{4\pi r^2} \frac{\lambda^2}{4\pi} \quad (3.14)
\]

Simplifying this equation and substituting frequency for wavelength results in Equation 3.15, where \( f \) is the frequency, \( c \) is the speed of light in a vacuum, and \( r \) is the distance between the transmitter and the receiver.

\[
L_{\text{fsl}} = \left( \frac{c}{4\pi fr} \right)^2 \quad (3.15)
\]

3.2.2 Atmospheric Loss

When travelling through the atmosphere, the molecules in the atmosphere will interfere with a signal, absorbing some of its energy and attenuating the signal power. Most of the attenuation due to atmospheric molecules is due to two components, oxygen gas and water vapor. The amount of attenuation the signal experiences is dependent on its frequency and the distance it travels through the atmosphere.
The chart in Figure 3.5 shows the amount of attenuation a signal will experience based on its frequency when the source is at zenith compared to a receiver on the ground at sea level. By using simple geometry, the approximate atmospheric loss at other elevation angles can be calculated using this chart.

![Figure 3.5: Atmospheric loss due to oxygen gas and water vapor [46]](image)

In order to calculate atmospheric losses more precisely, the Institute of Electrical and Electronics Engineers (IEEE) also has a standard available in [30]. This text contains a detailed method for calculating atmospheric losses, and takes into account the typical local weather at the receiver’s position, the elevation angle of the transmitter with respect to the receiver, and the required availability to determine the loss that should be assumed in the link budget. For lower frequencies (below 4 GHz), it is typically sufficient to assume a small loss; usually 0.5 to 1 dB of attenuation is more than adequate. For higher frequencies the attenuation can grow much larger, particularly due to the effects of water vapor and rain fading, and so it is more often calculated instead of assuming a value.

**Faraday Rotation**

When a electromagnetic wave is linearly polarized, the free electrons in the ionosphere can cause the polarization ellipse (or vector in this case) to rotate as it travels through the atmosphere [31]. This effect is known as Faraday Rotation, and is a result of right-handed and left-handed circularly polarized waves propagating at slightly different speeds through a magnetic field. By the superposition principle, any linearly polarized wave can be broken down into two circular components, one left-handed and one right-handed. The difference in propagation between these two components results in a rotation of the polarization when they are recombined [31]. For a wave travelling from the earth into space, the strength of this magnetic field is dependent on the number of free electrons in the ionosphere, also called the Total Electron Count (TEC). The TEC is normally measured in TEC units, equivalent to $10^{16}$ electrons per meter squared, and is affected by a number of factors including the time of day, sunlight intensity, and the sunspot cycle. The rotation of the wave is also determined by its frequency,
with higher frequencies being less affected. Equation 3.16 describes the angle of rotation $\Omega$ of a linearly polarized wave travelling through the ionosphere [31], where $TEC$ is the total electron content, $B$ is a model of the Earth’s magnetic field, $\lambda$ is the signal wavelength, $c$ is the speed of light in vacuum, $\theta$ is the angle between the direction of propagation and the magnetic field, and $\alpha$ is the angle between the direction of propagation and zenith.

$$\Omega = E_{TEC} \cdot B \lambda^2 \cdot \frac{2.365 \cdot 10^4}{c^2} \cdot \frac{\cos \theta}{\cos \alpha}$$ (3.16)

For L-band, the value of the Faraday rotation is relatively small, no more than $12^\circ$ at zenith during a solar cycle maximum [31]. For communication links with lower frequencies (and therefore higher wavelengths), Faraday rotation will become larger and therefore more significant of a problem with regards to maintaining correct polarization matching between transmitting and receiving antennas.

3.3 Receiver

The sensitivity of a receiver is the lowest received signal power at which the receiver can decode messages with a specified bit error rate (BER). The acceptable bit error rate for small satellite telemetry and command systems is typically on the order of 1 bit error in every 100,000 bits sent, but it will vary based on the link’s application and data packet size, and may be smaller or larger. By convention, a bit error rate of 1 in 100,000 is often expressed as a BER of $10^{-5}$. In order to make sure that the specified maximum BER requirement for a link is met, the signal to noise ratio of the signal at the detector must exceed a minimum required value, which varies based on the modulation scheme.

A receiver subsystem will always have both an antenna to collect and transfer RF power, and electronics to filter, amplify, and demodulate analog RF signals into data. In some cases an additional combiner, filter or low-noise amplifier (LNA) will also be added between the antenna(s) and the receiver in order to improve the sensitivity or protect it from damage or out of band noise. The goal of the receiver section of a link budget is to determine the receiver’s Gain/Temperature factor, also called its $G/T$. Like the $EIRP$ in the transmit section, this term expresses the effective contribution of all of the receive chain components to the ratio of signal to noise at the point of analog to digital conversion.

3.3.1 Receive Antenna

The receiving antenna works in the same manner as a transmitting antenna but in reverse, where radio waves induce a current in the antenna, which travels through a feed-line into the receiver, where it is amplified and decoded. As information was provided about antenna gain and bandwidth in Section 3.1.3, that information will not be presented again for receiver antennas, as they function in the same manner. Two factors that are specific to receive antennas, polarization loss and antenna temperature, are covered below.

Polarization Loss

Polarization loss arises due to the polarization mismatch between an incoming signal and a receiving antenna. Depending on the scale of the mismatch between the signal and the receive antenna, the resulting power loss can be significant. In order to calculate the mismatch loss, the general form of the polarization ellipse (Equation 3.10) is rather unwieldy. For the degenerate polarization cases, calculating
the loss between them is simple. Linearly polarized signals received by a linearly polarized antenna are attenuated depending on the offset angle $\Psi$ between their polarization vectors, using Equation 3.17. By this equation, the loss $L_{pol}$ is minimized when the angle $\Psi = 0^\circ$, and maximized when the angle $\Psi = 90^\circ$.

$$L_{pol} = \cos^2 \Psi$$

(3.17)

When using a circularly polarized antenna to receive a linearly polarized signal, the loss calculation is a bit different. As described in Section 3.16, a linearly polarized wave can be broken down by the superposition principle into two equivalent components, one right-hand circularly polarized, and one left-hand circularly polarized. A circularly polarized antenna receiving a circularly polarized signal of the same handedness will have no loss, while receiving a signal of opposite handedness will result in an infinite loss. This means that only the power of the signal that is polarized in the same handedness as the receiving antenna will be absorbed, with the other half of the power being lost.

These two cases are useful, but if a signal or receive antenna is not perfectly circularly or linearly polarized, the loss is more difficult to calculate. To do so, another term must be introduced. The Axial Ratio $R_A$ is the ratio between the major and minor axes of the polarization ellipse, and is a measurement of how elliptical the polarization is. An axial ratio of 1 means that the major and minor axes of the polarization ellipse are equal, and it becomes circular, while if the ratio is infinite, one of the axes is zero, and the polarization is linear. $\Psi$ is defined similarly to how it is used in Equation 3.17, but is the angle between the major axes of the ellipses rather than the polarization vectors. Equation 3.18 is a more general form, and can be used to calculate the loss between any two polarization ellipses, even those of the special degenerate cases. Subscript $tx$ denotes terms belonging to the transmitted signal, while subscript $rx$ denotes those belonging to the antenna.

$$L_{pol} = \frac{1}{2} + \frac{1}{2}(4R_A - t x R_A - r x + (1 - R_A - t x^2)(1 - R_A - r x^2) \cos 2\Psi}{(1 + R_A - t x^2)(1 + R_A - r x^2)}$$

(3.18)

It should be noted that even though the polarization loss is discussed as part of the receive antenna section, in a link budget it is typically considered along with the geometric losses such as free-space path loss. This is because it is not a resistive loss, and therefore doesn’t contribute to the noise power in the receiver.

**Receive Antenna Temperature**

Any receiving antenna on a spacecraft will pick up noise in addition to any desired signals, and it is important to quantify how this noise source contributes to the noise power in the receiver when modelling a communication system. The value of antenna noise temperature is typically represented by the black-body temperature of the thermal radiation emitted by the earth in Kelvin (290 K) for non-directional spacecraft antennas. The black body temperature used is dependent on the part of the Earth that the spacecraft’s antenna gain pattern is focused toward, with land typically having a higher temperature than the ocean [46].

An antenna’s effective temperature can be found by summing the product of the antenna gain and the black body temperature over the unit sphere, Equation 3.19 [41]. A directional antenna with a highly directional gain pattern pointed at space will have an antenna temperature of 4 K, and the same antenna pointed at the Earth will have an antenna temperature between 160 K and 290 K [46], depending on the target location on the Earth. The noise power received by an antenna can then be calculated using
Equation 3.20, where $k$ is Boltzmann’s constant and $B$ is the bandwidth over which the noise is received by the antenna [41].

$$T_{ant} = \frac{1}{4\pi} \int_0^\pi \int_0^{2\pi} G(\theta, \phi) T(\theta, \phi) d\phi d\theta$$

(3.19)

$$N_{ant} = kBT_{ant}$$

(3.20)

### 3.3.2 Receiver Electronics

By the time a signal transmitted from the earth reaches a satellite in space, it’s strength is typically weak enough that it can’t be detected directly. In order to detect these weak signals, the receiver must use a series of amplifiers and filters in order to increase the signal strength and reduce the amount of noise in the signal, respectively.

Figure 3.6: The CanX-7 ADS-B receiver’s receive chain

Figure 3.6 shows the components on the CanX-7 receiver board that make up the receive chain. There are three types of components in this receive chain: low noise amplifiers (LNAs), band pass filters, and a logarithmic power detector. The detector is the final stage of the analog signal chain before analog to digital conversion, and performs the conversion from the carrier frequency to baseband, before the resulting signal is sent to an FPGA for sampling. Band-pass filters remove out-of-band noise from the signal by suppressing frequencies above and below the desired frequency band, but also reduces the level of the signal of interest. A low-noise amplifier (LNA) amplifies the signal to increase the signal power, but also amplifies any noise that is present.

Proper placement of components in the receive chain is needed to obtain the best performance from a receiver. When designing a receive chain, both the total gain from the components and the amount of noise introduced into the signal by the components must be taken into consideration. Calculating total gain can be done easily, by summing the total gains and losses from each component. When added to the expected power level at the receiver, this value needs to be higher that the detection threshold of the detector and demodulator in order for conversion to baseband. The receive chain must also be designed such that the level of the noise introduced by the components does not exceed the level required for the demodulator to be able to distinguish between the desired signal and background noise. This is often
done by using several amplifiers and filters to increase power and filter out noise in stages. Ensuring that the signal to noise ratio is sufficiently large requires more mathematical effort than simple summation.

**Thermal Noise Floor**

Any signal other than the one that is desired can be considered noise, but the primary focus of this section will be on thermal noise. Thermal noise encompasses any undesirable signal generated in an active component by free moving electrons or absorbed from black bodies via the antenna. The thermal noise floor is the base value of thermal noise in a system. The thermal noise power $N$ in a given bandwidth can be quantified by using Equation 3.21. In this equation, $k$ is Boltzmann’s constant, $1.38 \times 10^{-23}$ J/K, $R_i$ and $R_o$ are the input and output impedances, respectively, and $T_0$ is a reference temperature, typically 290 K, the IEEE standard temperature for noise figure measurement.

$$N = 4kT_0B \frac{R_oR_i}{(R_i + R_o)^2}$$

(3.21)

This equation can be simplified if, as is typical in most radio systems, the input and output impedances $R_i$ and $R_o$ are equal. This causes the impedance terms and the constant 4 to cancel out, reducing the equation to Equation 3.22, exactly the same as 3.20. The bandwidth $B$, however, is not the bandwidth of the antenna, but the bandwidth of the frequencies amplified by the component. Typically the bandwidth of the last filter in the receive chain is used in this equation.

$$N = kT_0B$$

(3.22)

Based on Equation 3.22, it can be seen that the thermal noise floor can be decreased by changing either the signal bandwidth or the component temperature. In the context of a small satellite radio, where size and mass are at a premium and thermal control is typically passive, the temperature of the active receiver components are not controllable to enough of a degree to make a significant difference to the level of thermal noise present. The bandwidth then becomes the most important factor for determining the thermal noise floor.

**Antenna and System Noise Temperature**

The Friis formula for noise temperature, Equation 3.23 [41] can be used to calculate the signal to noise ratio degradation caused by the receive chain. The resulting value of noise temperature, along with the antenna gain, can be used to calculate the $G/T$ of the receiver. The Friis formula for noise temperature sums the noise temperature of the first active component with the noise temperature of the second component divided by the gain of the first component, and then with the noise temperature of the third component divided by the gains of the first two components multiplied together, and so on until the noise temperatures of each component have been taken into account. If a component is a passive resistive attenuator rather than an active component, it’s gain is equal to the resistive loss it causes, while it’s noise temperature is calculated by Equation 3.27. It can be seen that the Friis equation weighs the noise factor of the first component in the chain more heavily than the other components, as all the other components’ noise contributions are divided by the gains of each of the preceding components. Because the noise contribution of the first component will be multiplied by the gain of each of the following components, its noise temperature has the largest effect on the noise temperature of the receiver as a
Rather than quoting a noise temperature, most component manufacturers will quote noise figure instead. Noise figure $F_{\text{noise}}$ is equivalent to the signal to noise ratio at the input of the amplifier divided by the signal to noise ratio at the output, and is usually quoted at a reference temperature of 290 K. Converting between noise figure and noise temperature is accomplished through the use of Equation 3.24.

$$T_{\text{noise}} = 290(F_{\text{noise}} - 1) \quad (3.24)$$

When using the Friis formula for noise, it is important to remember that the component noise temperatures, noise figures, and gain values ($T_x$, $F_x$, and $G_x$, respectively, where $x$ is the component number in the chain) must be in linear units. If the logarithmic values (with units of dB) are used in the equation, the result will be incorrect. While all equations in this chapter have so far used linear units, it is more common to see values converted to the logarithmic scale before being factored into the link budget. Conversion from logarithmic ($x$) to linear ($y$) units can be done using equation 3.25.

$$y = 10^{x/10} \quad (3.25)$$

As the noise figure of a component can never be a negative, the use of amplification in the signal chain will always result in the total signal to noise ratio being reduced relative to when it is received by the antenna. By using a high-gain, low-noise amplifier early in the signal chain after reception, however, the signal to noise ratio can be kept as high as possible. For this reason, an external LNA is sometimes used in order to amplify the signal before being fed into the receiver to minimize the noise temperature contribution of the feed harness $T_{\text{feed}}$. If an external LNA is used, Equation 3.26 can be used to find the noise factor of the total LNA and receiver system.

$$T_{\text{chain}} = T_{\text{LNA}} + \frac{T_{\text{receiver}}}{G_{\text{LNA}}} \quad (3.26)$$

The feed harness is the transmission line, typically a coaxial cable, that connects the antenna to the receiver. Even though the feed harness should be designed to minimize losses, it still acts as an attenuator for the signal, and as such has a contribution to the noise temperature. The formula for the noise temperature of an attenuator is given in Equation 3.27 [41]. In this equation, $T_{\text{physical}}$ is the actual physical temperature of the component, while $L$ is the loss caused by the attenuator (i.e., for a 6 dB loss, $L$ is 0.25). This equation can also be used to calculate the contribution of filters, which are essentially frequency-selective attenuators.

$$T_{\text{attenuator}} = T_{\text{physical}} \frac{1 - L}{L} \quad (3.27)$$

With the noise temperature of each individual component identified, they can then be added together. The equation for the noise temperature of a system is equal to Equation 3.28, where $T_{\text{feed}}$ is the noise temperature of the feed harness between the antenna and first component in the receive chain.

\[ T_{\text{receiver}} = T_1 + \frac{T_2}{G_1} + \frac{T_3}{G_1G_2} + \ldots + \frac{T_n}{G_1G_2\ldots G_{n-1}} \quad (3.23) \]
With the noise temperature of the whole receiver subsystem, the G/T can be calculated. To do so, it is simply a matter of multiplying the antenna gain and any feed losses together, and dividing by the noise temperature of the system. This results in a single figure of merit that defines the performance of the receiver. This number can also be used to define the noise power in the receiver system using Equation 3.29, where the noise bandwidth $B$ is that of the last filter in the receive chain and $k$ is Boltzmann’s Constant.

$$N_{sys} = kT_{sys}B$$  \hspace{1cm} (3.29)

Sometimes a figure of merit defining the receiver performance, $\frac{G}{T}$ is derived from the receiver terms. This can be calculated using Equation 3.30.

$$\frac{G}{T} = \frac{G_{rx} \cdot L_{feed-rx}}{T_{sys}}$$  \hspace{1cm} (3.30)

### 3.4 Link Margin Calculation

The final step of completing the link budget is compiling the various terms and calculating the link margin. To do this, it is often simpler to use the logarithmic form of the terms found in previous sections. This can be done using Equation 3.31, the inverse of Equation 3.25. In this section, all terms will use their logarithmic values rather than their linear ones. This allows for representations of large gains and losses (such as free space loss) in a more clear and effective manner. It also means that rather than multiplying terms together, they can simply be added and subtracted from one another.

$$x = 10 \log_{10} y$$  \hspace{1cm} (3.31)

With $P_{eirp}$, whose components are calculated in Section 3.1, the losses due to the environment and polarization calculated in Sections 3.2 and 3.3.1, and the receiver antenna gain and feed harness loss discussed in Section 3.3, the signal level at the receiver input can be calculated using the Equations 3.32 through 3.35. It should be noted that even though losses are being added in these equations, when converted from a linear form to a logarithmic form they should be inherently negative numbers. For example, if the loss from a hypothetical filter $L_{filter} = 0.9$, this means that 90% of the power is transmitted through the filter, and 10% is attenuated. When converted to a logarithmic form, $L_{filter}$ would then be equal to -0.46 dB.

$$P_{eirp} = P_{tx} + L_{feed-tx} + G_{tx}$$  \hspace{1cm} (3.32)

$$L_{env} = L_{atm} + L_{fsl}$$  \hspace{1cm} (3.33)

$$G_{rec} = L_{pol} + G_{rx} + L_{feed-rx}$$  \hspace{1cm} (3.34)
\[ P_{\text{sig}} = P_{\text{eirp}} + L_{\text{env}} + G_{\text{rec}} \]  

(3.35)

The difference between the signal power \( P_{\text{sig}} \) and the noise power computed in Section 3.3.2 is then referred to as the carrier-to-noise ratio \( \frac{C}{N} \). Based on the type of modulation being used, there is a theoretical bit-error rate value that can be achieved for each value of \( \frac{C}{N} \). The carrier-to-noise ratio required to maintain the desired bit-error rate is then referred to as \( \frac{C}{N}_{\text{min}} \). Sometimes an additional term, implementation loss (\( L_{\text{imp}} \)), is also added to account for the difference between a particular demodulator’s ability to successfully decode bits at a particular \( \frac{C}{N} \) and the theoretical performance limit of the modulation being used [40]. Equation 3.36 is used to calculate the carrier-to-noise ratio of a received signal. Equation 3.37 is an expanded version of this equation, with terms grouped how they will often appear in a link budget.

\[ C_N = P_{\text{sig}} - N_{\text{sys}} \]  

(3.36)

\[ C_N = P_{\text{eirp}} + (L_{\text{atm}} + L_{\text{fsl}} + L_{\text{pol}}) + \frac{G}{T} - 10 \log_{10} k - 10 \log_{10} B \]  

(3.37)

The link margin is then defined by Equation 3.38 as the difference between the input carrier-to-noise ratio and the required carrier-to-noise ratio, inclusive of any implementation losses. Requirements for this margin will vary depending on the stage of design the calculation is being performed during, but between 3 and 6 dB is desired in most communication systems.

\[ M_{\text{link}} = \frac{C}{N} - \frac{C}{N}_{\text{min}} - L_{\text{imp}} \]  

(3.38)

### 3.4.1 Bit Energy to Noise Power Spectral Density Ratio

Sometimes rather than using \( \frac{C}{N} \), link margin is calculated using the bit energy to noise power spectral density ratio \( \frac{E_b}{N_0} \). This measurement is a normalized signal to noise ratio defined as the energy per bit divided by the noise power in one Hertz of bandwidth, and is a convenient way of representing the minimum required signal to noise ratio to maintain a link at a given bit-error rate using different modulations, independent of the modulation’s effect on a signal’s occupied bandwidth. \( \frac{E_b}{N_0} \) can be calculated from the carrier-to-noise ratio \( \frac{C}{N} \), the noise bandwidth \( B \) (in units of Hz), and the bit rate of the signal \( R_b \). The carrier to noise power spectral density ratio \( \frac{C}{N_0} \) is a bandwidth-independent carrier to noise measurement, found using Equation 3.39, that is calculated in an intermediate step before finding a signal’s \( \frac{E_b}{N_0} \) [33].

\[ \frac{C}{N_0} = \frac{C}{N} + 10 \log_{10} B \]  

(3.39)

After finding \( \frac{C}{N_0} \), it is divided by the bit rate of the modulated signal \( R_b \) to get \( \frac{E_b}{N_0} \). When using logarithmic units, \( R_b \) is subtracted from \( \frac{C}{N_0} \) rather than being divided by, as in Equation 3.40.

\[ \frac{E_b}{N_0} = \frac{C}{N_0} - 10 \log_{10} R_b \]  

(3.40)

It is common to see link margin calculated using \( \frac{E_b}{N_0} \), using Equation 3.38 but replacing the \( \frac{C}{N} \) terms with the equivalent \( \frac{E_b}{N_0} \) and minimum required \( \frac{E_b}{N_0} \). This method is accurate when the noise bandwidth
of a system is similar to the received signal’s occupied bandwidth. When the noise bandwidth is larger, as is the case for the link budget shown in Figure 3.1, using the carrier to noise ratio $\frac{C}{N}$ gives a more accurate result [12].
Chapter 4

Coverage Simulation

Typically when analyzing communication links, a link budget can be used in order to ensure that an acceptable margin is maintained based on the required parameters of the system. This is relatively simple to do when looking at a point-to-point system. The transmitter power and receiver thermal noise terms are based on the performance of the components used, and the transmit and receive antenna gains and environment and geometry losses can be estimated using the expected worst-case values where the link is required to function.

CanX-7 has some important system-level concerns which make this analysis method less representative of the ADS-B payload’s performance. The ADS-B air traffic management system is made up of a distributed network of transmitters (aircraft), the locations of which are not known and cannot be known a priori. The objective of the payload receiver is also not to ensure constant communication with a location over the ground for as long of a period as possible, as would be the goal of a telemetry and command receiver. The attitude control system used on CanX-7 is also designed to track only the Earth’s local magnetic field (LMF). As such, the spacecraft will rotate during its orbit in a consistent manner, but one that doesn’t keep its antenna pointed in a particular direction with respect to the Earth’s surface.

Based on these factors, a different analysis method is required in order to evaluate the expected performance of the payload while on orbit. The analysis method devised takes the spacecraft’s antenna pattern, attitude, receiver performance, and orbit into account, and results in a plot showing the swath coverage of the earth by the receiver. The end goal behind the coverage analysis is to determine how well the payload is expected to function, and how large an area the receiver will be able to cover during its operational passes. All calculations in this chapter were performed using MATLAB to both simplify the coding process and take advantage of mapping features present in pre-existing toolboxes.

4.1 Single Position Coverage Simulation

Three scenarios are examined in this analysis: the decoding performance of the receiver at several points during an orbit, a simulation of the coverage area during one recording pass similar to what would be performed during the operational phase of the mission, and a whole-orbit coverage simulation.

In the first scenario, the coverage is calculated for the visible part of the earth at a specific true anomaly. This true anomaly determines the Earth’s local magnetic field vector, calculated assuming a
simple dipole model for the Earth’s magnetic field, which then determines the spacecraft attitude, with the ADS-B antenna boresight tracking the local magnetic field vector. The output of this scenario is a map of the area on the Earth’s surface below the spacecraft with contours representing the expected message decoding percentage, essentially the inverse packet error rate, from aircraft transmitting within the areas denoted by the contours.

Expanding and rearranging Equation 3.35 from the previous chapter results in Equation 4.1. In this equation, the terms have been rearranged such that those dependent on the geometry between the spacecraft receiver and an aircraft transmitter are on the left, and those which are independent of it are on the right. This equation forms the basis of the simulations performed in this chapter.

\[
P_{\text{sig}} = (G_{tx} + L_{atm} + L_{fsl} + L_{pol} + G_{rx}) + (P_{tx} + L_{feed-tx} + L_{feed-rx})
\]  (4.1)

The simulation method described in this chapter requires the variables in this equation to be calculated and the equation to be solved for each aircraft position analyzed, which must be done tens of thousands of times for each true anomaly point analyzed (the precise number dependent on the required map fidelity). In order to minimize the calculation time and memory usage, the equation is broken down into constant terms and geometrically dependent terms. The constant terms are only calculated once, at the beginning of each simulation, and the geometrically dependent terms are calculated for each point as needed. As all simulations were performed in MATLAB, matrices were used to store all data for the simulations to allow for fast calculations without the use of loops. To further simplify the analysis, some additional assumptions were made about the satellite’s orbit.

The spacecraft was assumed to be in a circular 670 km polar orbit passing directly over the geographic poles, an inclination of 90°. In reality, most spacecraft launched into polar orbits will be slightly inclined in order to be sun-synchronous. The inclination of a sun-synchronous polar orbit at this altitude, however, is around 98°, only an 8° difference [46]. In addition, the orbit will still be near-circular regardless of the inclination, and so the swath area calculation will not differ between the assumed orbit and a sun-synchronous one, it will just be at a different inclination.

The earth was also assumed to be a perfect sphere, rather than an ellipsoid with an equatorial axis 21 km longer than its polar axis. The mean value of 6,371 km was assumed to be the radius of the Earth. In reality, with the spacecraft in a 670 km orbit, this means that an aircraft directly beneath the spacecraft at the poles is 14 km further away than in the simulation, while an aircraft directly below the satellite at the equator would be 7 km closer. This corresponds to a change in the received signal level of -0.2 dB at the poles or +0.1 dB at the equator, both relatively small changes. An aircraft not directly below the spacecraft would have its relative distance to the satellite affected even less, so the assumption was reasonable.

### 4.1.1 Link Equation Positionally Independent Terms

In order to run the calculations needed to perform the simulation, Equation 4.1 is broken down into two parts. First, all of the terms independent of the position and orientation of the transmitters and receiver are calculated and combined in order to reduce the number of redundant calculations which need to be performed during the simulation. These terms are defined as being positionally independent. Second, each of the terms dependent on the aircraft position relative to the spacecraft and the spacecraft attitude are calculated individually for all of the surface positions visible to the satellite. These terms
are defined as being positionally dependent.

**Transmitter Power**

Both the transmitter power and transmission and cable losses are consistent regardless of geometry. The ADS-B standard does not require transmitters to have a specific output power, and instead specifies a range of allowable values, from 125 to 500 Watts. There is no way to know in-situ what the transmitter power of an aircraft is however, and so a value must be chosen to represent all aircraft. The purpose of the analysis is twofold; to determine whether or not the ADS-B receiver will be able to pick up aircraft from orbit, and also to try to estimate the performance. Since the aircraft the CanX-7 payload is interested in detecting while flying over the North Atlantic are typically large commercial aircraft, they were assumed to have transmitter powers of 500 Watts.

**Cable Losses**

Losses from the cables connecting the transmitter to the transmitting antenna are assumed to be small at 0.5 dB, and are not dependent on the geometric position of the aircraft relative to the spacecraft. Receiver cable losses are measured to be 0.3 dB, and are similarly not affected by the aircraft’s geometric position.

**Polarization Loss**

Polarization loss comes from a mismatch between the polarization of the transmitting and receiving antennas. For the initial simulation, it was assumed that the receiving antenna was right-handed circularly polarized (RHCP). In the real system, the antenna will be close to but not perfectly circularly polarized, and it will have some axial ratio higher than zero. This will result in higher or lower losses for certain aircraft positions, dependent on the final axial ratio of the receive antenna. The effect of Faraday Rotation is also unpredictable, and as such it is difficult to calculate the expected polarization vector of any input signals. For this analysis, the loss due to polarization mismatch was assumed to be 3 dB, the level that would be expected from a system with a perfectly linear and perfectly circular antenna. This number is a good-enough approximation for determining the area that the payload covers.

### 4.1.2 Link Equation Positionally Dependent Terms

The remainder of the terms in the coverage simulation are dependent on either the position of the transmitting aircraft relative to the spacecraft, or the spacecraft attitude (which is itself dependent on the true anomaly). These terms must be calculated for each aircraft position, and so it is important for the calculation process to be as efficient as possible.

**Aircraft Position Definition**

The terms $G_{tx}$, $L_{fsl}$, and $L_{atm}$ in Equation 3.38 are dependent on the aircraft position relative to the spacecraft. For simplicity, it is assumed that all aircraft will be in a state of steady, level flight, at an altitude of 11 km. This assumption is reasonable for most commercial aircraft, which spend a majority of their time in steady flight at cruising altitudes.
The simplest way of defining aircraft positions relative to the spacecraft is by using elevation and azimuthal angles with respect to a nadir pointing orbit fixed reference frame centered on the spacecraft. Figure 4.1 displays how the angles defining aircraft locations are defined with respect to the spacecraft location. The elevation angle $\theta_v$ is defined as the angle formed by a vector pointing in the spacecraft nadir direction and a vector from the spacecraft pointing to a hypothetical aircraft below. The azimuthal angle $\gamma$ is defined as the angle formed by a projection of the aircraft vector into the plane normal to the nadir direction and the spacecraft velocity vector. The true anomaly defining the spacecraft location within the orbit is represented as $\theta_s$.

In order to reduce the total calculation time, it is important to use a number of aircraft points in the coverage simulation small enough to calculate quickly, yet large enough such that the fidelity of the result is not significantly reduced. The azimuthal angle $\gamma$ must range from 0 to 360 degrees, and the elevation angle must range from 0 degrees (nadir) to the horizon to compute the signal level of all the locations with line of sight to the spacecraft. Equation 4.2 is used to calculate the elevation angle of an aircraft at the horizon, where $r_e$ is the radius of Earth, $h_{sat}$ is the orbital altitude of the satellite, and $h_{ac}$ is the altitude of an aircraft, all in kilometers. Once the limits are known, the interval between points is chosen. Due to the data structure which will be used to perform and store results of calculations, explained in section 4.1.2, this portion of the calculation is actually fairly efficient, and so a reasonably tight spacing of one degree was used for both elevation and azimuth. The result of the maximum angle equation for CanX-7’s orbital altitude of 670 km is 66 degrees, and so the total number of points to be tested is $67 \times 361 = 24,187$ points.

$$\theta_{v-max} = \sin^{-1}\left(\frac{r_e + h_{ac}}{r_e + h_{sat}}\right)$$  (4.2)

**Calculation Matrices**

With these two angles defined, MATLAB is used to generate a meshgrid of all the combined elevation and azimuthal angles that define aircraft locations. Each element location in the matrices represents a potential aircraft location, with one matrix storing elevation angles, and one storing azimuthal angles. Each matrix has a number of rows corresponding to the number of elevation angles (67), and a number
of columns corresponding to the number of azimuthal angles (361) to be checked. The elements in the meshgrid matrices correspond to $\theta_v$ values in the elevation angle matrix (4.3), and $\gamma$ values in the azimuthal angle matrix (4.4).

\[
\begin{bmatrix}
0 & 0 & 0 & \ldots & 0 \\
1 & 1 & 1 & \ldots & 1 \\
2 & 2 & 2 & \ldots & 2 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
66 & 66 & 66 & 66 & 66
\end{bmatrix}
\]

(4.3)

\[
\begin{bmatrix}
0 & 1 & 2 & \ldots & 360 \\
0 & 1 & 2 & \ldots & 360 \\
0 & 1 & 2 & \ldots & 360 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 1 & 2 & \ldots & 360
\end{bmatrix}
\]

(4.4)

All of the position dependent terms in the analysis use this format to store data. To calculate each term, equations are constructed as normal, with azimuth and elevation angles as inputs. Then, the entire matrix is input into each equation to calculate all of the values of the equation for each aircraft position simultaneously. The resulting output matrix then stores all of the results in a manner consistent with the original position definitions.

**Aircraft Antenna Gain**

ADS-B is broadcast from quarter-wavelength monopole antennas, mounted on either the bottom or both the top and bottom of an aircraft. In this analysis, it is expected that messages will only be received from the top antenna [25], and so the transmitting antenna is assumed to be top-mounted. A monopole antenna has a linear polarization and a gain pattern symmetrical around the axis of polarization. This means that the aircraft direction doesn’t factor into the expected antenna gain. The aircraft antenna gain equation is then one-dimensional, only depending on the vertical angle from the aircraft to the satellite.

Equation 4.5 describes the aircraft antenna gain at a point dependent on the elevation angle $\theta_{ac}$ from the zenith direction of the transmitting aircraft, where 3.28 is the directivity of a quarter-wavelength monopole antenna [25]. This angle is that which is formed from the zenith direction to the vector between the aircraft and the spacecraft. In order to use this in simulation, a transformation is applied to convert the angle $\theta_v$ into the aircraft fixed frame of reference, using Equation 4.6. Figure 4.2 is provided as a reference for the problem geometry.

\[
G_{tx} = 10 \log_{10} \left(3.28 \frac{\cos^2 \left( \frac{\pi}{2} \cos(\theta_{ac}) \right)}{\sin^2 \theta_{ac}} \right)
\]

(4.5)

\[
\theta_{ac} = \pi - \sin^{-1} \left( \frac{r_c + h_{sat}}{r_c + h_{ac}} \sin \theta_v \right)
\]

(4.6)
Atmospheric and Free Space Loss

The calculation of atmospheric loss is described in detail in Section 3.2.2. At 1090 MHz, this loss is quite small and is estimated by using the chart in Figure 3.5 as 0.03 dB at zenith. To calculate atmospheric loss in directions other than zenith, the value can be calculated by dividing the value by the cosine of $\theta_{ac}$, Equation 4.7. This equation is an approximation only, but works adequately for values of $\theta_{ac} < 85^\circ$.

$$L_{atm} = \frac{L_{zenith}}{\cos \theta_{ac}} \quad (4.7)$$

Free space loss is dependent on the frequency of the signal being transmitted and the distance between the aircraft and the spacecraft. Since the frequency that ADS-B is transmitted at is fixed at 1090 MHz, the only term in the equation that varies based on the aircraft position is the distance $r$ from the transmitter to the receiver. This distance is calculated using Equation 4.8, derived using the Law of Sines, where $r_e$ is the radius of the Earth and $h_{ac}$ is the altitude of an aircraft. The free space loss equation (Equation 3.15) can then be converted to its logarithmic form, Equation 4.9. Equation 4.9 is only dependent on the input of constants and the angle $\theta_{ac}$, which is itself only dependent on constants and $\theta_v$.

$$r = (r_e + h_{ac}) \frac{\sin(\theta_{ac} - \theta_v)}{\sin \theta_v} \quad (4.8)$$

$$L_{fsl} = 20 \log_{10} \left( \frac{c}{4f \pi r} \right) \quad (4.9)$$

4.1.3 Spacecraft Antenna Gain

The spacecraft antenna gain $G_{rx}$ is the most difficult term in Equation 4.1 to calculate. Unlike the other positionally dependent terms, the equations for which have been dependent on a single variable and independent of the spacecraft attitude, the spacecraft antenna gain in the direction of each aircraft position is dependent on both the spacecraft attitude and the position of the transmitting aircraft in
both azimuth and elevation in relation to the spacecraft.

**Spacecraft Attitude**

The spacecraft is assumed to be tracking the local magnetic field vector with the boresight of the ADS-B antenna, the nominal attitude control system (ACS) mode during operations. Initial ACS simulation [17] showed that the spacecraft was expected to stabilize in this manner during payload operations, with the long axis of the spacecraft fixed in the orbital plane. These assumptions allow the description of the spacecraft attitude to be simplified down to a single variable, the angle from the orbit velocity vector to the spacecraft antenna boresight vector, hereafter referred to as the antenna pointing angle $\beta$.

**Antenna Gain Determination**

The first step in determining the spacecraft antenna gain is to find the antenna pointing angle $\beta$ using a lookup table of the values output from the attitude model. This defines the rotation between the spacecraft body-fixed frame and the orbit fixed frame. Then, the elevation and azimuth angles defining aircraft positions from Section 4.1.2 are transformed into the spacecraft body frame. This is done by converting the elevation and azimuth angles into a set of three-dimensional unit vectors and transforming them using a single-axis rotation into the spacecraft body frame. Additional transformations can then be applied if the gain pattern is defined in a reference frame other than the spacecraft body frame. The unit vectors are then converted back to elevation and azimuth angles $\theta_{ant}$ and $\gamma_{ant}$, defined now in the reference frame of the antenna.

An antenna gain pattern defined in the antenna reference frame using spherical coordinates is then imported and used to determine the spacecraft antenna gain for each of the aircraft locations. When performing simulations during the early design stages, a hypothetical cardioid antenna pattern was used to define the gain. In the simulation results presented later in Sections 4.1 and 4.3.1, the antenna gain pattern used is that measured during the antenna pattern test. With the antenna pattern and aircraft positions defined in the same reference frame, two-dimensional cubic interpolation is used to determine $G_{rx}$. All of the calculated terms can then be combined to determine the input power at the receiver $P_{sig}$.

**4.1.4 Message Decoding Percentage**

With the received power at the input to the receiver calculated, it is desirable to convert this into a more meaningful measurement indicative of payload performance. Estimated message decoding percentage was chosen as the figure of merit here. Message decoding percentage is defined in this case as the inverse of packet error rate, where a packet error rate of 30% equates to a decoding percentage of 70%. A function can be defined that relates decoding percentage to the input signal level in order to calculate the expected decoding percentage. In early simulations this was done by using a best-fit curve matched to sensitivity test performance data, while the simulation results presented in this chapter will use a curve fitted to electromagnetic compatibility test data.

**Contour Generation**

To generate contours on the earth showing the payload coverage, the aircraft position data $(\theta_v, \gamma)$, which has so far been defined with respect to the satellite’s orbit-fixed reference frame, is converted into
earth-fixed Cartesian coordinates. This is done by calculating the satellite’s position based on its true anomaly $\theta_s$, the spacecraft altitude, and an input longitude, and defining aircraft positions based on distances and directions relative to the spacecraft. The Cartesian coordinates are then converted into global latitude and longitude values, which can be plotted on a map more easily. The last step before a plot can be generated is the creation of a contour map of the data. This is done using MATLAB’s built-in contour generation function.

4.2 Single Spacecraft Position Results

Once the single position simulation is set up, inputs are chosen for the true anomaly, spacecraft longitude, spacecraft altitude, and aircraft altitude. For the simulations presented here, the starting spacecraft longitude is $40^\circ$ West, the spacecraft altitude is 670 km, and the aircraft altitude is 11 km. The plots in Figure 4.3 show the expected performance with the spacecraft at several true anomaly values. In the figure, each contour represents a 10% increase in expected decoding percentage, with the outermost contour indicating 0% to 10% decoding percentage, the second representing 10% to 20%, and so on.

At the equator ($0^\circ$), the spacecraft antenna is pointed above the horizon. In this case, even though the antenna beamwidth is fairly wide, most of the antenna’s gain is not pointed towards the earth, and thus the area of coverage is fairly low. The point at which the maximum area is covered by the ADS-B receiver is when the spacecraft is at 40 degrees true anomaly. In the plot in the bottom right, it can be seen that there is a large hole in the middle of the contours, despite the spacecraft antenna being pointed almost directly at this location. This is due to a null in the aircraft antenna pattern at zenith. The maximum received signal power will be seen by the spacecraft when Equation 4.1 is maximized. By removing the terms assumed to be constant, this can be reduced to the cost function $J$ listed in Equation 4.10. With the spacecraft near an aircraft’s zenith point, $J$ is dominated by the low transmitter gain $G_{tx}$, despite $G_{rx}$, $L_{atm}$, and $L_{fsl}$ being maximized. This means that reception there is poor, with no aircraft able to be detected.

$$J = G_{tx} + L_{atm} + L_{fsl} + G_{rx}$$  \hspace{1cm} (4.10)

4.3 Coverage Swath

With the ability to generate a single position coverage map, doing so for a given swath is relatively straightforward. The process used to do this is outlined below.

1. Choose starting and ending true anomaly values
2. Generate single position coverage maps at each of the true anomaly values
3. Split contours when they cross the $+/- 180^\circ$ longitude line into separate polygons
4. Combine the contour polygons generated by the point coverage simulations at each decoding percentage level into a single polygon
5. Plot the new contours and shade their interiors
Figure 4.3: Coverage with the spacecraft at various true anomaly values (clockwise, from top-left): 0°, 20°, 80°, 40°

4.3.1 Coverage Swath Results

The result of the coverage swath calculation is shown in Figure 4.4. The map on the left shows the area covered during a typical recording pass, while the map on the right shows the area covered were the payload to be left running over an entire orbit.

4.4 Conclusion

The designed simulation technique shows that the CanX-7 ADS-B payload will be able to detect aircraft from space. It will achieve expected peak decoding percentages of greater than 60%, and be able to cover a large swath of the Earth at greater than 50% decoding percentage. The simulation also shows that optimal performance is achieved in the middle latitudes between 30 and 60 degrees North, matching well with the North Atlantic target area. With the magnetorquer polarities reversed, this same swath area could also be covered over the Southern Hemisphere, with optimal performance in the latitudes above
Chapter 4. Coverage Simulation

Figure 4.4: Coverage swath during a typical 18 minute recording pass (left) and over a full spacecraft orbit (right)

Australia and New Zealand, the payload’s secondary targets.
Chapter 5

Antenna Design and Testing

When the prototype ADS-B payload was first received from RMCC in January of 2015, one of the tasks that needed to be completed before the payload was flight-ready was the design of a new antenna. The prototype antenna concept was for a circularly polarized patch antenna with low gain and narrow bandwidth. It was intended that a low thermal noise bandwidth would compensate for the low gain of the design to maintain high sensitivity to ADS-B signals [44]. However, with so little tolerance for error during the manufacturing process, there had been issues getting the center frequency correct.

![Prototype Antenna vs Final SFL Designed Antenna](image)

Figure 5.1: The prototype antenna (left), next to the final SFL designed antenna (right)

5.1 Modification of the Prototype Antenna Design

Before any modifications were made to the antenna design, it was important to identify key requirements that would drive design decisions. The antenna needed to be capable of picking up messages from aircraft over a wide area in order to maximize the number of aircraft detected on a single pass. This was critical during the design phase, as at that point the payload was only intended to operate for two minutes per orbit. At such a low duty cycle, the spacecraft would risk missing large numbers of aircraft if the beamwidth was too narrow.

The antenna also needed to be capable of operating at a wide range of pointing angles. The attitude profile of CanX-7, and thus the boresight of the ADS-B antenna, is aligned with Earth’s magnetic field
vector during payload operations, due to CanX-7’s magnetic attitude control system. In most cases, the boresight will not be pointed in an optimal direction. In addition to maximizing the area that the antenna covers, a wide beamwidth also ensures good performance even when the antenna is not pointed optimally.

Polarization mismatch between transmitted signals and receiving antennas can be one of the largest sources of loss in a link unless properly mitigated. Since ADS-B transmitters are linearly (vertically) polarized, the polarization mismatch loss can be up to 100% if the polarization vector of the receive antenna is also linear and is misaligned. In addition, the electrons in the ionosphere can cause rotation of the polarization vector of a transmitted ADS-B signal at L-Band frequencies, meaning that even if the receive antenna is aligned with the transmitter polarization vector at the time of transmission, once the signal propagates to the receiver, the alignment may change. See Sections 3.2.2 and 3.3.1 for background on polarization and how it affects a communications link.

It was also important to design an antenna that could be built with realistic tolerances during the manufacturing process. Simulation results are not always perfectly reflected by real-world performance, therefore the design must be able to maintain adequate performance over a range of manufacturing process and material imposed tolerances. Since the antenna is being designed to be compatible with an existing spacecraft structure and not the other way around, the antenna must also meet any mechanical requirements imposed upon it by the spacecraft bus, and perform adequately in the expected thermal environment.

5.1.1 Patch vs. Monopole Antenna

The prototype ADS-B antenna was a circularly polarized patch, with a narrow bandwidth and low gain. It was decided that before any work was done to design a new antenna, alternative types of antennas would also be considered against requirements in order to ensure that a patch was the appropriate choice. Various types of antennas are discussed in Section 3.1.3, with several presented in Table 5.1 Most of the types of antennas were discarded due to either performance or size limitations, but it was decided to compare a quarter-wavelength monopole and patch antenna to assess the benefits and drawbacks of the both options.

<table>
<thead>
<tr>
<th>Antenna Type</th>
<th>Reason for Rejection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dipole</td>
<td>Deployable required, low peak gain, can’t be directed towards Earth</td>
</tr>
<tr>
<td>Helical</td>
<td>Mechanically complex, deployable required</td>
</tr>
<tr>
<td>Horn</td>
<td>Too large at ADS-B frequency, incompatible with spacecraft deployer</td>
</tr>
<tr>
<td>Log-periodic</td>
<td>Too large at ADS-B frequency, incompatible with spacecraft deployer</td>
</tr>
<tr>
<td>Parabolic</td>
<td>Too large at ADS-B frequency, incompatible with spacecraft deployer</td>
</tr>
</tbody>
</table>

A patch antenna sits flush with the spacecraft body panel it’s mounted on, and has a semi-directional cardioid gain pattern with a wide beamwidth between 70 and 100 degrees [29]. A monopole would sit perpendicular to the body panel it is mounted on, and has an axial omni-directional gain pattern concentrated in a ring shape at some elevation angle away from its ground plane, depending on its electrical length and the size of the ground plane. Simulations were performed using the technique
presented in Chapter 4 using both a monopole and patch antenna. The monopole gain was determined using Equation 4.5, and the patch antenna gain was determined using Equation 5.1 for a cardioid with a peak gain of 5 dBi, where $a$ is the peak gain in linear units. The patterns were assumed to be symmetrical about the boresight for the patch antenna, and the null direction for the monopole, with $\theta$ in both equations equal to the angle measured from the boresight/null vector. The monopole antenna was assumed to be linearly polarized, while the patch was assumed to be circularly polarized.

$$G_{\text{cardioid}} = 10 \times \log_{10} \left( a (1 + \cos \theta) \right)$$  \hspace{1cm} (5.1)

![Figure 5.2: Line of sight coverage of the Earth’s surface for monopole (left) and cardioid patch (right) antennas, with the spacecraft located at (0,0)](image)

The simulations in Figure 5.2 show the expected link margin for an aircraft located on the Earth’s surface below the spacecraft, assuming the spacecraft is in a 650 km orbit. The reference frame is defined based on the spacecraft position, with the Z-axis in the zenith direction, the Y-axis in the direction of the velocity vector and the X-axis in the direction of the orbit normal. Dark blue areas outside of the lowest contour line represent those below the decoding threshold, where messages are not expected to be received. Ultimately, it was decided to continue using a patch antenna for the ADS-B payload for several reasons.

- Mechanically, the patch is less complex. A monopole would need to be deployable to avoid interfering with the spacecraft deployment mechanism.

- The patch covers a larger overall area due to it’s circular polarization and higher gain when pointed towards the earth. The 3 dB loss due to circular polarization is constant regardless of the incoming signal polarization.

- The linear polarization of a monopole results in a smaller overall area being covered by the antenna. It also suffers from the effects of Faraday Rotation, which will rotate the polarization vectors of ADS-B signals as they pass through the ionosphere.
5.2 Patch Antenna Design

Several aspects of the prototype antenna design were carried over into the SFL design. The antenna has a probe-fed, circular radiating element designed to emit circular polarization. The probe feed is preferable to a microstrip feed as it allows the cable to the LNA to connect to the back of the antenna and be routed internally. A circular radiating element was used over a square one, as a circular patch will be smaller than an equivalent square patch designed for a specific frequency [11], and an area only 75 mm square was available on the spacecraft for mounting. The design uses a single feed, with a cutout in the center to induce circular polarization, rather than relying on a two port solution. Using two ports to induce circular polarization is simpler than using one, but requires the use of a combiner before demodulation.

Unlike the prototype design, gain was prioritized over narrow bandwidth. This was done by changing the substrate material to one with a higher dielectric constant and increasing its thickness significantly. The shape of the cutout in the center of the radiating element was also altered to be more asymmetrical and achieve better polarization circularity.

5.2.1 Antenna Design Simulation

During the design process, much of the work was done in tandem with another S.F.L. student, Paris Ang. An initial design for the antenna was created by Paris Ang, at which point an iterative design process was undertaken with the assistance of the author in order to achieve the desired performance parameters. This section only gives a brief overview of the design process, as the main focus of this chapter is on the antenna requirements, the selection of the antenna type, and of the testing and acceptance of the manufactured antenna.

Performance requirements for the patch antenna, are presented in Table 5.2. In order to optimize the antenna design, several of the design variables were varied using the parametric analysis feature available in the HFSS antenna simulation software. The parameters that were varied are presented in Table 5.3. Figure 5.3 shows the geometry of the antenna with respect to the variables used to describe it.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>75 mm x 75 mm</td>
</tr>
<tr>
<td>Center Frequency ($f_c$)</td>
<td>1090 MHz</td>
</tr>
<tr>
<td>Peak Gain</td>
<td>$\geq 4$ dBi</td>
</tr>
<tr>
<td>Half-Power Beamwidth</td>
<td>$\geq 90^\circ$</td>
</tr>
<tr>
<td>Reflection Coefficient at $f_c$ ($S_{11}$)</td>
<td>$\leq -10$ dB</td>
</tr>
<tr>
<td>Impedance Bandwidth ($S_{11} \leq -10$ dB)</td>
<td>$\geq 10$ MHz</td>
</tr>
<tr>
<td>Axial Ratio Beamwidth ($AR \leq 3$ dB)</td>
<td>$\geq 90^\circ$</td>
</tr>
</tbody>
</table>

The antenna went through four distinct iterations prior to manufacturing, not including the prototype antenna. Each of the iterations is referred to by a designator (I1, I2, etc.) for clarity. The design of I1, which was performed analytically, was rejected due to poor gain performance. The peak gain of this design was -0.22 dBiC, more than 4 dBiC less than the design requirement.
In the second iteration of the design, I2, the material was changed from Rogers TMM6 to TMM4, with a lower dielectric constant. The substrate thickness was also increased, resulting in a more efficient antenna with higher gain over a wider bandwidth. Because of the lower dielectric constant of the substrate in I2, the patch size had to be increased, almost to the same size as the underlying substrate dimensions. When the antenna was placed on the spacecraft bus, there was a significant shift in the antenna’s axial ratio of more than 20 dB. It was thought that because the edge of the patch was so close to the edge of the substrate, the spacecraft bus may have disrupted the fringing fields at the radiating element’s edge leading to the performance shift on the spacecraft bus.

A third iteration was performed as an experiment to test this hypothesis. I3 was designed using Rogers TMM6, which allowed the size of the patch to be reduced, and an expanded 80 mm x 80 mm substrate was used in the simulation. Though this meant that design I3 would be non-compliant
with the size requirement, it was not an issue as it was only intended to determine the cause of the frequency shift. The design of I3 showed the same shift in center frequency as that seen in I2 when placed on the spacecraft bus. As the axial ratio was still not compliant with requirements and there was not enough time to investigate and quantify the cause of the shift, it was decided that all future design iterations would be performed with the antenna on the spacecraft bus. This choice significantly increased computation time for each iteration, but removed the need to determine precisely how the spacecraft bus was affecting the antenna performance.

For the fourth iteration, the antenna was taken down two diverging paths. Design I4a was intended to keep the design parameters as similar as possible to I3, but with the size of the substrate reduced in order to meet the size requirement. Design I4b used a thicker substrate, longer, thinner cross-slot arms, and optimized the rotation of the cross-slot by trying $\kappa$ angles other than $45^\circ$, which until this point had not been altered. The performance parameters for I4a and I4b are shown in Table 5.4. I4b was selected as the flight design, as the gain performance was weighed more heavily than the slightly lower axial ratio and larger beamwidth of I4a, due to the tight requirements imposed by the link budget. I4a only barely met the 4 dBi peak gain requirement, and if the performance of a manufactured antenna was any worse than in the simulation, the resulting antenna would not meet requirements.

<table>
<thead>
<tr>
<th>Performance Parameter</th>
<th>I4a</th>
<th>I4b</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak Gain</td>
<td>4.0 dBi</td>
<td>4.63 dBi</td>
</tr>
<tr>
<td>Half-Power Beamwidth</td>
<td>147°</td>
<td>142°</td>
</tr>
<tr>
<td>Reflection Coefficient at $f_c$ ($S_{11}$)</td>
<td>-18.2 dB</td>
<td>-11.2 dB</td>
</tr>
<tr>
<td>Impedance Bandwidth ($S_{11} \leq -10$ dB)</td>
<td>20 MHz</td>
<td>35.5 MHz</td>
</tr>
<tr>
<td>Axial Ratio Beamwidth ($AR \leq 3$ dB)</td>
<td>127°</td>
<td>121°</td>
</tr>
</tbody>
</table>

For additional details on the design iterations, including the initial analytical calculations and design and performance parameters of each of the variants, see [11].

5.2.2 Initial Design Manufacturing

Once the design process was completed, the antenna was sent to a local printed circuit board shop for manufacturing. Two antennas were made, one for flight and another designated as a spare unit. The manufactured I4b design can be seen in Figure 5.4.

Antenna Acceptance Testing

Acceptance testing for the antenna was simple, and consisted of checking the impedance bandwidth and impedance values only, before a full antenna pattern test was performed. Thermal acceptance of the antenna was not performed, as simulations showed that the worst-case hot and cold temperatures expected to be experienced by the spacecraft would not cause a significant shift in the dielectric properties of the antenna substrate, and thus the antenna performance would remain stable.

The manufactured antennas were attached to a spare spacecraft structure to take acceptance measurements, using a vector network analyzer. A picture of the test results of the I4b antenna are shown
in Figure 5.5. The test results showed that both antennas had issues with their center frequencies being lower than designed. It was determined that the $S_{11}$ curves of the antennas were shifted due to an incorrect dielectric value used during the design of iterations I1-I4b.

When the antenna was first designed, the process dielectric constant of the substrate material was used as the dielectric constant in simulations, as this value was stored by default in HFSS for the material and it was assumed that this value was correct. However, another value called the design dielectric coefficient is actually recommended by the material manufacturer as the best approximation of true dielectric constant for use in computer assisted engineering applications [16]. For many materials, the design and process dielectric constants are very close or even the same, but for Rogers TMM6, the selected substrate material for design I4b, the process dielectric constant is 6.0 and the design dielectric constant is 6.3 [39]. When altering the simulation of the I4b antenna to use the design dielectric coefficient of 6.3, the $S_{11}$ curve very closely matched that measured during the tests, as seen in Figure 5.6.
Figure 5.6: Simulated S11 curve of antenna I4b using the design dielectric constant (left) and the measured curve from antenna I4b (right)

5.2.3 Redesign Using Design Dielectric Constant

Correcting the simulation to use the design dielectric constant, the antenna was redesigned in a fifth and final iteration, starting from iteration I4b and using the same heuristic tuning process as the first four iterations. The resulting design parameters are shown in Table 5.5, and the performance metrics are shown in Table 5.6. An image of the gain pattern is displayed in Figure 5.7.

Table 5.5: Design Parameters of the flight antenna, iteration I5

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Design Variable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Substrate Design Dielectric Constant ($\epsilon_r$)</td>
<td>6.35</td>
</tr>
<tr>
<td>Substrate Thickness ($t_{sub}$)</td>
<td>5.08 mm</td>
</tr>
<tr>
<td>Patch Diameter (2$r$)</td>
<td>61 mm</td>
</tr>
<tr>
<td>Probe Distance from Center ($p$)</td>
<td>8.81 mm</td>
</tr>
<tr>
<td>Cross-Slot Rotation Angle ($\kappa$)</td>
<td>45°</td>
</tr>
<tr>
<td>Arm X1 Length ($l_{x1}$)</td>
<td>18.4 mm</td>
</tr>
<tr>
<td>Arm X1 Thickness ($t_{x1}$)</td>
<td>1.91 mm</td>
</tr>
<tr>
<td>Arm X2 Length ($l_{x2}$)</td>
<td>9.3 mm</td>
</tr>
<tr>
<td>Arm X2 Thickness ($t_{x2}$)</td>
<td>0.5 mm</td>
</tr>
</tbody>
</table>

Table 5.6: Performance metrics of the design iteration I5

<table>
<thead>
<tr>
<th>Performance Parameter</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peak Gain</td>
<td>4.36 dBi</td>
</tr>
<tr>
<td>Half-Power Beamwidth</td>
<td>95°</td>
</tr>
<tr>
<td>Reflection Coefficient at $f_c$ ($S_{11}$)</td>
<td>-17.6 dB</td>
</tr>
<tr>
<td>Impedance Bandwidth ($S_{11} \leq -10$ dB)</td>
<td>31.7 MHz</td>
</tr>
<tr>
<td>Axial Ratio Beamwidth ($AR \leq 3$ dB)</td>
<td>127°</td>
</tr>
</tbody>
</table>
Flight Antenna Manufacturing

With the design finalized, four antennas were made: one designated for flight, one designated as a flight spare, and two engineering models. To test the antennas, they were placed on a spare spacecraft structure and plugged into a virtual network analyzer (VNA) to measure the center frequency, bandwidth, and S-parameters, as before. Each of the four antennas were tested for $S_{11}$ performance, with the two best performing antennas selected as the flight and spare units. The measurements from these antennas met requirements for center frequency and impedance bandwidth, so pattern testing was performed on the flight antenna to determine whether it met gain performance requirements.

5.3 Antenna Pattern Testing

Pattern testing was performed on the manufactured antenna to verify that its gain, beamwidth, and axial ratio met requirements. To perform this test, the antenna along with a spare satellite structure (also called a tinsat) were taken to the University of Toronto Electromagnetics Lab, to be tested in their anechoic chamber. In the section, all axial ratio and gain values listed use logarithmic units, unless otherwise specified.

5.3.1 Setup

In preparation for the test, the antenna was mounted to the spare spacecraft structure, along with the two S-band downlink antennas and four UHF uplink monopoles used by the telemetry and command system. In addition, the area on the +Y panel where the drag sails would normally be located was covered with aluminized Kapton sheeting in order to mimic the effect of the sail cartridges being mounted in the flight structure. The S-band antennas were connected together via a splitter, with a coaxial cable fed
out of the structure through one of the holes in the -Z panel. The ADS-B antenna was also connected directly via a coaxial cable fed through the -Z panel, but the UHF antennas were left unconnected. As they would not be used during the testing, they did not need be connected to the equipment, but it was important to account for how they would affect the pattern of the ADS-B antenna and its isolation from the S-band downlink antennas.

To perform the pattern test measurements, several pieces of equipment were used. A signal generator was used to transmit a carrier wave at the ADS-B frequency from a source horn antenna, and a network analyzer was used to measure the signal power and phase from the antenna under test. A coupler was placed on the signal generator output and run through a coaxial cable outside of the anechoic chamber.
Figure 5.9: The ADS-B antenna and two S-band downlink antennas are connected internally via coaxial cable and fed outside the structure.

Figure 5.10: The CanX-7 pattern test structure with deployables folded (left) and unfolded (right) to act as a reference for the network analyzer. A schematic showing how the equipment is configured is shown in Figure 5.11.

Figure 5.11: Antenna pattern testing test equipment schematic.
For the antenna gain to be calculated, the gain of the source antenna, the free-space loss, and cable losses had to be calibrated out of the measurement. The correction factor added to the gain of each measurement can be calculated using Equation 5.2, where $C_F$ is the correction applied to each magnitude measurement, $G_{\text{Ref}}$ is the gain of the source antenna, $L_{\text{2Antennas}}$ is the signal power measured via the network analyzer when calibrating with two source antennas, and $L_{\text{FeedCables}}$ is the loss due to the any additional cables needed to connect the ADS-B antenna to the test setup.

$$C_F = G_{\text{Ref}} - L_{\text{2Antennas}} - L_{\text{FeedCables}}$$  \hspace{1cm} (5.2)

Calibrations were performed before running any testing in order to measure the gain of the source antenna at 1090 MHz, as well as to categorize the free space loss between the source and receiver, and to calibrate out the losses experienced by the signal as it passed from the signal generator, through the coupler, and into the network analyzer reference point. Figure 5.12 shows the source horn antenna used for the testing. The laser level within the antenna was used to ensure proper alignment, and was removed prior to any test being run.

**5.3.2 Test Plan**

The antenna pattern was measured by placing the spacecraft in four orientations, each offset by 45 degrees in the X axis, and rotating the entire spacecraft 360 degrees to calculate the gain pattern in a two dimensional pattern cut. Two special foam stands were made to support the spacecraft in these orientations, shown in figure 5.13. The foam used is sturdy enough to hold the spacecraft, but allows most RF radiation to pass through.

With the spacecraft in each of the four orientations, the foam stand and spacecraft are placed on a rotating pedestal. Two sets of measurements were made in each orientation, one each with the source horn antenna oriented in the horizontal and vertical directions, to measure the antennas response to horizontally and vertically polarized radiation. During each measurement, the pedestal with the spacecraft rotates a full 360°, with both the magnitude and phase of the received signal measured at one degree intervals for a range of frequencies centered on 1090 MHz. With the magnitude and phase measured in both vertical and horizontal polarizations in all of the test orientations, a full $4\pi$ steradian
antenna pattern was interpolated in all polarizations.

5.3.3 Results

In order to calculate the measured gain from the network analyzer in each of the orientations, Equation 5.3 is used to calculate the horizontal and vertical gain measurements. $A_{H,V}$ represents the measured amplitude of the signal by the vector network analyzer in one of the two polarizations measured in decibels, while $C_F$ is the correction factor calculated using Equation 5.2.

\[ G_{H,V}(dB) = A_{H,V} + C_F \]  \hspace{1cm} (5.3)

With the corrected gain calculated in each polarization, equations 5.4 and 5.5 can be used to calculate the left handed and right handed circular polarized gains of the antenna [10]. In these equations, $G_H$ and $G_V$ are the linear forms of the gain $G_{H,V}$ calculated in Equation 5.3, calculated using the measured horizontal and vertical gains from the test. $P_H$ and $P_V$ are the respective phase measurements. Axial ratio can be calculated using the LHCP and RHCP gain values via Equation 5.6.

\[ G_{LHCP}(dB) = 10 \log_{10}([G_H \cos(P_H) + G_V \sin(P_V)]^2 + [G_H \cos(P_H) - G_V \sin(P_V)]^2) - 3 \]  \hspace{1cm} (5.4)

\[ G_{RHCP}(dB) = 10 \log_{10}([G_H \cos(P_H) - G_V \sin(P_V)]^2 + [G_H \cos(P_H) + G_V \sin(P_V)]^2) - 3 \]  \hspace{1cm} (5.5)
\[ R_A = 20 \log_{10} \left( \frac{10^{\frac{G_{RHCP}}{20}} + 10^{\frac{G_{LHCP}}{20}}}{10^{\frac{G_{RHCP}}{20}} - 10^{\frac{G_{LHCP}}{20}}} \right) \]  

(5.6)

**Calculated Antenna Pattern**

A polar axial ratio plot and RHCP gain plot were generated for each of the four spacecraft orientations tested. Only the first orientation, L-000, is shown here for conciseness.

![Calculated Antenna Pattern Image]

**Figure 5.14: Measured axial ratio (left) and gain (right) of the I5 antenna**

Pattern testing indicated that the performance of the manufactured antenna was similar to that of the designed antenna, with both a higher peak gain and axial ratio than seen in simulation. The higher axial ratio was something of a concern, as the antenna as measured was non-compliant with the axial ratio requirement. The measured performance parameters of which are included below in Table 5.7.

<table>
<thead>
<tr>
<th>Performance Parameter</th>
<th>Pattern Tested Antenna, Orientation L-000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>75 mm x 75 mm</td>
</tr>
<tr>
<td>Center Frequency</td>
<td>1090 MHz</td>
</tr>
<tr>
<td>Peak Gain</td>
<td>4.6 dBi</td>
</tr>
<tr>
<td>Half-Power Beamwidth</td>
<td>95°</td>
</tr>
<tr>
<td>Reflection Coefficient at ( f_c ) (( S_{11} ))</td>
<td>-18.77 dB</td>
</tr>
<tr>
<td>Impedance Bandwidth (( S_{11} \leq -10 ) dB)</td>
<td>35 MHz</td>
</tr>
</tbody>
</table>

**5.4 Conclusion**

Based on pattern testing and reflectance measurements, the simulated and manufactured antennas met all of the requirements listed in Table 5.8, with the exception of axial ratio beamwidth.
Table 5.8: ADS-B patch antenna design requirements

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Requirement</th>
<th>Compliant?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>75 mm x 75 mm</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Center Frequency ($f_c$)</td>
<td>1090 MHz</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Peak Gain</td>
<td>$\geq$ 4 dBi</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Half-Power Beamwidth</td>
<td>$\geq$ 90°</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Reflection Coefficient at $f_c$ ($S_{11}$)</td>
<td>$\leq$ -10 dB</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Impedance Bandwidth ($S_{11} \leq -10$ dB)</td>
<td>$\geq$ 10 MHz</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>Axial Ratio Beamwidth ($AR \leq 3$ dB)</td>
<td>$\geq$ 90°</td>
<td>NOT COMPLIANT</td>
</tr>
</tbody>
</table>

With an axial ratio of 5 dB at boresight, the manufactured antenna has a higher axial ratio than was expected based on simulations. This means that if used to receive linearly polarized ADS-B signals, rather than being a constant 3 dB, the polarization loss of the manufactured antenna will vary based on the aircraft/spacecraft geometry. In certain cases, the loss will be greater than that of a perfectly circularly polarized antenna, while in others the loss will be lower. It was decided that this trade was acceptable, and would not impact the ability of the payload to perform its technology demonstration mission, so the antenna was accepted for flight.
Chapter 6

Payload Control Software

When the prototype ADS-B receiver payload was delivered to SFL in January of 2015, it came with a basic suite of software that would allow it to boot, initialize itself, record ADS-B data, and transfer that data to the CanX-7 OBC. While this software would have worked to complete the payload’s primary mission objective, added functionality was desired in order to ensure the safety of the payload, improve its performance, and increase its flexibility.

With these goals, it was decided to expand the functionality of the prototype software rather than writing new software from scratch. This reduced development time, as it meant that several of the key features of the original software could be re-used in the flight software.

6.1 Prototype Software and Requirements Generation

While functional for the purpose of achieving the mission, the prototype software was designed to function in a linear manner, offered little flexibility for operations, and was lacking many of the features typically required of SFL software. Figure 6.1 shows a flowchart describing the prototype software.

Several deficiencies were identified with the prototype software, which needed to be addressed in the flight version:
1. The software operated in a linear manner, with the data file from the previous operation automatically overwritten when starting up.

2. The payload had no telemetry reporting functionality.

3. The software would not respond to commands while recording ADS-B data.

4. The software did not contain any functionality for updates while on orbit.

5. No event or command log was generated by the payload which could be downloaded in the case of a crash or error.

6. The payload did not synchronize its local clock with the spacecraft clock, so received messages were not time-tagged.

7. The recording time was static and could not be changed once set, reducing the flexibility of the payload.

### 6.1.1 Software Requirements Generation

With the prototype software serving as a starting point, several requirements were developed for the flight payload software. These requirements were designed with the aim of increasing the flexibility, reliability, and capability of the receiver. These requirements were not formal mission requirements, but were defined in order to provide direction for the software design.

1. The payload computer shall communicate with the spacecraft computer via the Nano-Satellite Protocol (NSP) [32].

2. The payload computer should implement all of the standard NSP commands, including Ping, Init, Peek, Poke, Bulk Read, Bulk Write, and Parameter Read/Write.

3. The payload computer shall be able to reply to commands at any point during operation.

4. The payload computer shall boot from an external storage device.

5. The payload computer should use the Angstrom Linux Operating System.

6. The external storage device shall contain two partitions, one of which shall be read-only once on orbit.

7. The read-only storage partition shall contain the operating system and bootloader, and the writable partition the application software, FPGA image, and data storage.

8. The payload computer shall be able to boot without accessing any files located on the writable storage partition.

9. The application software, receiver software, and FPGA image shall be updatable while the spacecraft is on orbit.

10. The receiver FPGA shall run and interface with the payload computer using commercially available receiver software.
11. The payload should maintain an internal clock accurate to within 1 second of the spacecraft clock for the purpose of time-tagging events and ADS-B data

12. The payload computer should keep a log of received commands and error/warning messages

13. An interface should be available for interacting with the operating system via terminal commands

14. The payload computer should be capable of being underclocked in order to reduce power consumption

15. Successfully decoded ADS-B messages shall be stored and time-tagged to within 1 second of the time of reception

16. ADS-B messages shall be automatically transferred to the spacecraft OBC at the end of each recording

17. The payload recording duration shall be configurable at the beginning of each recording

18. The payload shall be capable of reporting basic telemetry to the spacecraft including state, uptime, and temperature

6.1.2 Software Architecture

The ADS-B receiver flown on CanX-7 is designed to work both mechanically and electrically with the embedded Linux computer used as the payload computer, and its manufacturer-included software is designed to be compatible with both the Angstrom Linux and Debian Linux operating systems. Since the prototype software was written on the Angstrom Linux distribution, it was decided that this operating system choice would be maintained in order to reuse as much of the original code as possible. Some significant changes would have to be made, however, in order to meet the requirements defined in Section 6.1.1. The flight version of the software that was written to control the payload is broken up into four distinct segments: the operating system, the pseudo-bootloader, the application, and the receiver software.

6.2 Software Development Testbed

To aid in the software development process, three receivers and payload computers were purchased at the beginning of the project. One set was designated as flight units, and underwent the full suite of hardware modifications described in Section 2.3.2. Another set was designated as flight spare unit. These were partially modified, with the receiver only having its GPS chip removed and the payload computer’s USB and ethernet ports left attached. The remaining receiver/computer pair was designated as an engineering model, with the only modification being the removal of the GPS receiver chip on the receiver.

The flight computer and flight receiver were stored away while the majority of software development took place using the spare and engineering model computers and receivers. The engineering model was powered using a 5 Volt DC power supply, and communicated with a laptop via a 3 Volt (CMOS level) UART to USB adapter and with the SFL internal network via ethernet. The UART connection was used for testing, as this is the interface used by the spacecraft computer for communicating with the
payload, while the ethernet connection was used to directly access the file system and for command via Linux terminal. The spare unit was placed on the CanX-7 flatsat, and was used for further development and testing in a more flight-like setup. The flatsat contains all of the spacecraft hardware mounted to a metal plate and connected together in a flight-like configuration, and is shown in Figure 6.2. This allows for debugging and software development with a full suite of spacecraft hardware, without actually having to assemble the satellite.

Figure 6.2: The ADS-B payload on the CanX-7 flatsat

6.3 Payload Computer Boot Sequence

When power is applied to the payload computer, as it is capable of booting from multiple sources, it checks each in turn for a bootable partition. The following locations are checked, in order:

1. Onboard eMMC flash storage (MMC1)
2. External microSD flash storage (MMC0)
3. Serial connection (UART0)
4. USB connection (USB0)

When a location is checked, the processor looks for a file called \textit{MLO} on a properly FAT formatted partition, which allows the processor to bring up the U-boot bootloader and begin loading the Angstrom Linux operating system. If the file is not found on a properly formatted partition, the processor moves on to the next boot location.

Software requirement 4 states that it must run off of an external microSD card, as opposed to the computer’s on-board flash memory. This was done for two reasons. First, it meant that a microSD card rated for industrial use could be purchased separately for the receiver. The higher operating and
storage temperature ranges of the industrial rated external card that was used meant that the risk of damaging the memory if the payload overheated for some reason was reduced. Additionally, booting from a removable memory card meant software development was far easier. A removable memory card of the same size as the flight memory card could be plugged into the EM payload for development, then removed and placed into the spare on the flatsat for testing, and then imaged and written to the flight card, all using existing software and simple memory card readers. The images could also be compressed and stored during development, making it simple to keep backup copies of the entirety of each software iteration, including changes made to the operating system.

To force the computer to boot from the microSD card, the \textit{MLO} file was removed from the internal eMMC memory, but the rest of the file system was left intact. It also would have also been possible to simply clear the partition of all files in order to force the computer to boot from the memory card. However, the internal memory was used to store a backup copy of the flight software as a potential way of restoring the file system if it became corrupted. This feature was never implemented due to complexity and time constraints, but a backup copy of the final code (minus the \textit{MLO} file), was still included on the internal memory for flight, just in case the feature was implemented in the future.

\subsection{Payload Computer File System}

The payload computer flash memory is divided into two partitions. The first, hereafter referred to as partition 0 (P0), contains the operating system and pseudo-bootloader. The second, referred to as partition 1 (P1), contains the application script, all of the receiver software, and data and log files. A Windows utility was used to create the partitions on the memory card, and then the standard receiver software package (including the Angstrom Linux operating system) was loaded onto the partition intended to be read-only, P0. When first booted from using the EM payload, the file system mounting table was modified to automatically mount P1 to a location on the file system. All of the receiver software was also moved from its default installation directory on P0 to a new folder on P1, with a symbolic link to the new directory replacing the original directory. This meant that for the purpose of performing updates to the receiver software during development, compatibility would be maintained with the manufacturer’s installers, while also keeping the files on P1 rather than P0, allowing them to be modified on-orbit if need be.

\subsection{Pseudo-Bootloader}

The pseudo-bootloader is so-called because while it is not a true bootloader, it does encapsulate much of the functionality of one with regard to the payload control software. It is stored on the read-only P0 partition, and contains basic functionality used to configure the payload computer serial ports, communicate with the spacecraft computer, respond to pings, read from and write to files, and to execute Linux terminal commands. The pseudo-bootloader itself is made up of four parts: a startup service, a management shell script, the pseudo-bootloader python script, and a watchdog trigger script.

The pseudo-bootloader forms the first segment of the flowchart shown in Figure 6.3, which describes how the software operates.
6.4.1 Software Management Script

At boot, a startup service launches a shell script that serves to manage all of the other components of the software, `OBC_cx7.sh`. The script is based on one included with the standard suite of receiver software, but with some significant changes to enable much of the functionality of the ADS-B payload. The script first verifies that compatible computer and receiver hardware versions are present, and then sets up two of the computer’s five UART serial lines to talk to the receiver. UART5 is used for communication with the receiver, including flashing the FPGA and sending ADS-B messages to the computer, while UART2 is used for communication with the receiver’s GPS chip. Even though the GPS receiver chip was removed from all of the receiver boards to reduce power consumption, the configuration of this serial line was included in the management script. As the receiver software was proprietary, the source code could not be viewed and there was no easy way to determine how it would deal with not being able to access this UART line. Since the serial line wasn’t needed for anything else, no change was made. The script then configures UART4 for communicating with the spacecraft OBC, and the on-board analog-to-digital converter for the temperature sensor. It should be noted that each UART line and the ADC are configured only if they are not configured already. This prevents conflicts if the startup service is restarted and the script is re-run after the initial boot.

With all of the communication hardware configured properly, the payload then loads two more scripts and runs them in new threads simultaneously. One is a shell script that manages a hardware watchdog timer on the receiver board, and the other is a python script that contains most of the functionality of the pseudo-bootloader.

Once the management script has loaded this python script, it will wait for the script to exit before
continuing. Once it does exit, the management script will load the FPGA image to the receiver board, initialize the receiver software, and load the ADS-B receiver application script. The application script is designed to run in a loop, but unlike the pseudo-bootloader script, there is no command designed to cause the script to complete so that the management script can continue to its next instruction, which would cause it to end. There are, however, some commands that will restart or stop the higher-level process that manages the startup service. These will be discussed in Section 6.6.

Receiver Hardware Watchdog

The receiver was originally designed to be used as an ADS-B data collection and feeding device for online aircraft tracking services. Because of this, the receiver is designed to be left on for long periods of time without requiring any human interaction. If the computer stops functioning properly to the point where the receiver software fails, then it is desirable in the original application for the computer to automatically restart itself, and load its default software configuration. To this end, a hardware watchdog timer was included that must be triggered once per minute or it will force a reboot of the payload computer. Normally this is done by the receiver software, but in order to meet Requirement 8, the payload must be able to boot into a stable configuration without it.

To get around this problem, a simple shell script was written to automatically trigger the hardware watchdog on the receiver without having to rely on the receiver software running. The script initializes general purpose input/output (GPIO) on pin 60, sets the value to 1 for one second, then sets the value to 0 for twenty nine seconds. This effectively cycles the voltage on the pin high once every thirty seconds, triggering the watchdog to restart its counter.

This watchdog triggering script is important, as it allows the payload to remain in its pseudo-bootloader state for the upload and re-writing of software without having to initialize the FPGA or load the receiver software.

6.4.2 Pseudo-Bootloader Program

The pseudo-bootloader program (PBP) is so called because it is not a true bootloader in the way that the bootloader on SFL spacecraft house-keeping computers is. Rather than loading an operating system, the PBP is designed to execute basic payload computer functions in a safe, non user-modifiable manner. Like all software that is loaded automatically on boot by the payload computer (i.e. without instruction from the spacecraft OBC), it is located on the read-only partition P0. Similar to the application script, the PBP is written in Python, a high-level language. Python 2.7 was chosen for this application primarily because it was what the prototype software was written in, and so did not have to be entirely written from scratch. Python also had several other advantages over other languages with respect to this project. It is a high-level language, and as such it was easy to learn and use, and short development time was prioritized over computational efficiency during this project. This is not to say that no care was paid to writing code that operated in an efficient manner, but rather that the inefficiency of the language itself was not considered as important as the speed of development using the language. The Beaglebone Black has a relatively powerful ARM Cortex-A8 processor, and as such the additional processor overhead needed to run Python scripts is far outweighed by the simplicity of the language and the ease of implementing advanced functions through reliable, pre-built packages.
Pseudo-Bootloader Initialization

When the PBP is first loaded, it initializes itself by loading any necessary packages and setting up logging handlers. Rather than simple print statements, using the logging package allows more versatility in how messages are stored; they can be appended with the time and level of urgency, and also stored in multiple locations or printed to the main console with one command, based on the definition of the handlers at program start. In the bootloader, as the program should not need to access files on the writeable partition P1 to function, messages are only printed to the system console rather than being stored in a file. After setting up logging, the PBP also defines its version number, checks its last modified date, and defines default locations for log files, write buffer file, and ADS-B data file. These definitions are stored as global variables so they do not need to be individually passed to functions that require them. Finally, the serial connection to the spacecraft OBC is opened and the main receive packet thread is started as a background thread. The script’s main thread then enters a continuous loop that constantly checks a global flag variable that indicates whether or not the script should continue. Each iteration checks whether the background thread is still active, and restarts it if it is not. This is to prevent unexpected events from triggering a collapse in the PBP, as if it were to crash all communication with the payload by the spacecraft would be impossible, and the payload would have to be power cycled.

Pseudo-Bootloader Commands

The background thread loaded during initialization handles all of the communication with the spacecraft OBC. The basic architecture for parsing NSP packets was re-used from the prototype software, but updated to use event logging and to fix a packet framing bug. The thread sequentially reads individual bytes from the serial connection with the OBC, storing the sequence in a buffer and replacing any escape sequences with their representative characters, until it either reaches a maximum packet length or it receives a frame-end character [32]. Once this happens, the packet buffer is checked for a valid CRC code. If one is not present, the packet buffer is cleared and the process begins again. If the CRC code is valid, the packet is sent to a parsing function.

As with the packet receive function, the NSP parsing function was partially reused from the prototype software, but a suite of new command functions were created to add additional functionality to the payload. The NSP parser first checks the destination address on the packet. If it doesn’t match 0x50, the address of the payload, the packet is dumped and a response is sent to the OBC indicating non-acknowledgement, also called a NACK. If the address is correct, then the command number is checked against a list of available commands. If the specified command number isn’t one of those available in the PBP, then another NACK is sent to the OBC.

If the command is recognized by the PBP, it then acts on the command. In the command byte of an NSP packet, the first three bits are called the P/F bit, the B bit, and the A bit, respectively, while the remaining five bits represent the command number [32]. The P/F bit indicates whether a response is requested by the OBC, the B bit is unused in this application, and the A bit indicates whether the packet is a command or a response. Table 6.1 shows the command list of the PBP, including the command number, the command name, any required or optional data, and a brief description of the functionality or response for each command. When the data field requires more than one input, they are listed sequentially.
Table 6.1: ADS-B payload command list when running the Pseudo-Bootloader Program

<table>
<thead>
<tr>
<th>Command</th>
<th>Required Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>N/A</td>
<td><strong>PING</strong> - Responds with the software version, build date and time, and payload clock time if set. Included in prototype software, but only returned the static string “PONG”.</td>
</tr>
<tr>
<td>0x02</td>
<td>N/A</td>
<td><strong>PAYSTATE</strong> - Commands the payload to respond with either an eight character string “BOOTLOAD” indicating the PBP is running, or an eight character string containing the operating status and application version if the application script is running.</td>
</tr>
<tr>
<td>0x03</td>
<td>N/A</td>
<td><strong>TEMP</strong> - Responds with the average of five readings taken from the temperature sensor in Celsius, in ASCII text format.</td>
</tr>
<tr>
<td>0x04</td>
<td>N/A</td>
<td><strong>FILESIZE</strong> - Responds with a little-endian integer specifying the size of the target file, with the ADS-B data file as the default target.</td>
</tr>
<tr>
<td>0x05</td>
<td>4-byte address, 1-byte number of packets, up to 32</td>
<td><strong>BULKREAD</strong> - Commands the payload to send a specified number of packets from the target file (default ADS-B data file), beginning at the four-byte location specified, over the serial link to the OBC. Included in prototype software.</td>
</tr>
<tr>
<td>0x06</td>
<td>4-byte address, string in ASCII hex format</td>
<td><strong>CODEWRIT</strong> - Commands the payload to write a string into the write buffer file at a location specified byte a four-byte address.</td>
</tr>
<tr>
<td>0x07</td>
<td>1-byte integer</td>
<td><strong>WRTQUERY</strong> - Handles a write log query from the OBC. 0x00 in the data field clears the write log and returns empty. An empty data field returns the 32 write log entries.</td>
</tr>
<tr>
<td>0x09</td>
<td>1-byte flag, string in ASCII hex format</td>
<td><strong>CREATEFL</strong> - Commands the payload to transfer the write buffer file to a new file with a name and location specified by the ASCII string. The flag is a hex byte indicating whether the write buffer file should be cleared after the transfer (0x00), cleared without transferring the file (0xff), or transferred without clearing the write buffer file (any other byte).</td>
</tr>
</tbody>
</table>


| 0x0a | String in ASCII hex format | **TERMINAL** - Commands the payload to execute a Linux terminal command. The terminal output (up to 256 characters) is sent in reply, along with a flag indicating whether the response is larger than 256 characters. The most recent full response for each command is stored in the terminal log file, which can be read out via the BULKREAD command. |
| 0x0b | 1-byte flag | **BOOT2CTL** - Flips the flag in the main thread required to keep it running. This causes the PBP to end, and the management script to proceed loading the FPGA image and launching the receiver software and the application script. |
| 0x0e | 1-byte flag | **TELMETRY** - Responds with the requested telemetry points, in order. Each bit in the flag byte represents one telemetry point, in the order [0,1,2,3,4,5,6,7]. The following points telemetry types are available: 0 - Temperature in degrees Celsius, 4-byte float; 1 - System state, 8-byte ASCII string; 2 - Free storage on partition P1 in bytes, 4-byte integer; 3 - ADS-B file size in bytes, 4-byte integer; 4 - Payload uptime in seconds, 4-Byte float; 5 - System clock time in seconds from J2000 epoch, 8-byte double. |
| 0x0f | N/A | **SHUTDOWN** - Commands the payload to safely shut down using the Linux shut down procedure. The payload responds with the static ASCII string “Shutting Down”. |
| 0x10 | String in ASCII hex format | **READTRGT** - Changes the target file for the BULKREAD and FILESIZE commands to that specified by the ASCII string. |

While an effort was made to implement many of the standard NSP commands in the PBP as per Requirement 2, some of the commands were not practical to implement. In particular, the Peek and Poke commands were not implemented. On most devices, the Peek command is used to read out a particular memory address, while Poke is used to insert a specified value into memory at a specific location. Implementing these commands was deemed too difficult for the limited benefit that they would provide, as ensuring compatibility with the Angstrom Linux OS would be very time consuming. The functionality of the Peek command was replaced by providing a basic interface to the Linux terminal through the PBP, and since most of the code that operates the ADS-B payload consists of relatively small scripts, it was determined that uploading entire new code files over the uplink would not be prohibitive from a data uplink throughput standpoint, so the Poke command was deemed to not be necessary.

### 6.5 Receiver Software Package

The receiver is packaged with control software that can be used for both high-level interaction and low-level control by allowing access to recorded ADS-B data via either network or serial connection.
When one of the two development testbed payloads was connected to the local network via ethernet, this graphical user interface (GUI) was a convenient way of checking the status of the receiver, editing settings, and verifying that messages were being received during testing.

The most important function of the receiver software with respect to the payload control software is managing the FPGA on the receiver and handling the distribution of received ADS-B messages in a manner that makes it easy to collect and store them via the application script.

One of the biggest advantages to using a commercial ADS-B receiver is that both this software and the FPGA image have already been written by the manufacturer to decode ADS-B messages. Developing FPGA software to handle the ADS-B signal processing would require far more time, effort, and a different skillset than was required to interface with the pre-existing software.

### 6.5.1 ADS-B Message Handling

During operation, the FPGA samples the output of the logarithmic detector at a rate of 16 MHz looking for an ADS-B message preamble, according to the manufacturer. Once a preamble is detected, the FPGA converts the next sequence of samples into data bits. Once each message is received it is sent over the UART5 serial link, where the receiver software running on the payload computer checks its CRC code and ensures that it is valid. In the prototype software, data was recorded directly from the serial port without checking the CRC codes to ensure that the frames were valid. In the flight implementation, the receiver software is configured to output messages with valid CRC codes over the network interface using the local loopback. This process is simple to implement in Python, and reduces the size of the data files to be downlinked by only storing messages that have been received without any bit-errors, one of which will render an entire 112 bit packet of data invalid.

Once the CRC code has been checked by the receiver software, it processes the message to store internally, and also passes the message over its ethernet connection in binary format. In this format, messages begin with the hex character 0x1a and an identifier character depending on the message type, with the true hex character 0x1a encoded as 0x1a 0x1a. The messages are distributed in the following formats based on the message type:

1. Mode-A/C message - [0x1a 0x31], 6-byte multilateration (MLAT) timestamp, 1-byte signal level, 2-byte data message
2. Mode-S short message - [0x1a 0x32], 6-byte MLAT timestamp, 1-byte signal level, 7-byte data message
3. Mode-S long message - [0x1a 0x33], 6-byte MLAT timestamp, 1-byte signal level, 14-byte data message

### 6.6 Application Software

The application software is designed in a very similar manner to the PBP, but also includes additional commands to interface with the receiver software. It is also written in Python, and follows a similar architecture with regards to parsing commands and interacting with the spacecraft OBC. This section focuses on the differences between the application script and the PBP, including logging, interaction with the receiver software and receiver board, and data storage and transfer.
6.6.1 Application Initialization

Once the application script is started, much of the initialization is similar to that of the PBP, but with some additional functionality enabled. An additional logging handler is enabled to event messages to a log file, the clock is synchronized with the spacecraft clock, and the processor is underclocked from 1 GHz to 600 MHz to save power. It is also important to emphasize that the application script is not a subprocess of the PBP. Before switching between the PBP and application scripts, the management script is waits for the PBP to end, and then launching the application script as a new process. This means that when the application script is started it needs to both restart the spacecraft communication serial connection and re-locate many of the data files used by the script.

Two logging handlers are enabled in the application script, one printing to the console and another that records all of the payload events to a log file. This log file is fixed to a maximum size of 262 kB, at which point it is re-designated as a backup log file, and a new file is created. The payload software will save the five most recent backup files in this manner. Once a sixth file is designated as a backup by the software, the oldest backup file is deleted. These files can be downloaded as necessary for diagnosing issues with the application software or the payload hardware.

To reduce the power consumption of the payload, the script also commands the payload CPU to underclock itself to 600 MHz rather than running at its maximum 1 GHz clock rate. This feature was very important to the operational capability and flexibility of the payload on orbit, as it reduces both its power consumption and heat generation. Extensive testing was performed before implementing this feature, to ensure the processor could still keep up with a high rate of incoming ADS-B messages. Originally it was intended to underclock the processor further to only 300 MHz, however it was found that this sometimes would cause incoming ADS-B messages to be back up in a buffer until they began getting dumped as new data came in, reducing the message decoding percentage.

Payload Computer Clock Synchronization

Time synchronization with the spacecraft OBC was implemented on the payload in order to enable timetagging of ADS-B files. To reduce the complexity of the mission and size of the spacecraft bus, CanX-7 did not include a GPS antenna or receiver to provide a sub-microsecond accuracy clock onboard. The spacecraft clock did, however, provide a relatively accurate clock with which to synchronize the payload. A simple method for obtaining the spacecraft clock time was implemented which required no modification of the spacecraft operating system software.

When the application script is initialized, the payload sends a ping command to the spacecraft, and waits for a response. When the spacecraft OBC receives a ping request, it responds with a standard ping response string. Unlike the PBP, the application script is configured to check the A-bit in each NSP packet it receives. In commands from the OBC, the A-bit is 0 (unchecked), but in replies it is 1 (checked). When the payload computer receives a ping message from the OBC with the A-bit checked, it knows to process the received ping response string rather than respond to a command from the OBC. This string contains information about the spacecraft state, it’s software version, and build date, and the clock time in seconds from the J2000 epoch. The payload computer clock is then set by converting the time in seconds from the J2000 epoch time into the standard UNIX time epoch of January 1 1970. If the ping response is successfully received by the payload, a flag is then tripped in the application script to indicate that the time has been synchronized successfully.
As the time obtained in this method only has precision to within one second, it is possible that using this method introduces a clock error of up to one second in the payload clock with respect to the spacecraft clock. This error is, however, on the same order of magnitude as the expected drift of the spacecraft clock. This method of time synchronization was determined to be sufficient for the payload, because any effort spent improving the accuracy of the synchronization method would have had only limited benefit to the end result message timetag accuracy.

### 6.6.2 Application Commands

Many of the commands used by the application script are similar to those used by the PBP, and are described in Table 6.1. Table 6.2 contains a list of commands that are either specific to the application script or differ in functionality between the two scripts.

<table>
<thead>
<tr>
<th>Command</th>
<th>Required Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x01</td>
<td>4-byte number of frames, 4-byte timeout, 4-byte recording mode flag, all little endian</td>
<td><strong>RCRD</strong> - Commands the payload to record ADS-B data from the receiver software via the network interface. The number of frames and timeout define when to end the recording, a 0 in either field will tell the payload to record an infinite number of frames or for an infinite time, respectively. The 4-byte flag defines what types of messages to record: long ADS-B frames (0x03 0x00 0x00 0x00), short ADS-B frames (0x02 0x00 0x00 0x00), or both (0x01 0x00 0x00 0x00).</td>
</tr>
<tr>
<td>0x08</td>
<td>N/A</td>
<td><strong>SRESTART</strong> - Commands the payload to restart the startup service that runs the management script described in Section 6.4.1. This kills the thread containing the management script and all child processes, including the application script. The management script is then started by the new instance of the startup service.</td>
</tr>
<tr>
<td>0x0b</td>
<td>N/A</td>
<td><strong>BOOT2CTL</strong> - Does nothing in the application script.</td>
</tr>
<tr>
<td>0x0c</td>
<td>N/A</td>
<td><strong>KILLRCRD</strong> - Commands the payload to end the current ADS-B recording early.</td>
</tr>
<tr>
<td>0x0d</td>
<td>4-byte integer, little endian</td>
<td><strong>CLOCKSET</strong> - Commands the payload to set the system clock to the 4-byte integer specified in the data field. The integer is the time in number of seconds since the J2000 epoch.</td>
</tr>
</tbody>
</table>
6.6.3 ADS-B Recording Thread

One of the biggest differences between the prototype software and the flight software is the way that ADS-B messages are recorded. To begin a recording, the RCRD command (0x01) must be sent to the payload, along with a data string containing a number of messages to record, a recording timeout, and a message type to filter for. The number of messages to record and the recording timeout define end conditions for the recording, each of which can be disabled by entering a 0 in their respective fields. The payload concept of operations was to record all ADS-B messages received from aircraft, regardless of their length or type. The option was provided to filter for only long frames or short frames in order to reduce data volume if there was ever any throughput backup once on orbit. Figure 6.4 is a flowchart describing how the ADS-B recording thread works.

Requirement 3 states that the payload computer must be able to reply to commands at any point during operation. While recording ADS-B data, the payload must still be able to carry out various housekeeping functions, in particular it must be able to respond to telemetry requests and commands to end the recording early. Python allows for simple multi-threaded execution of functions through the use of the `threading` package. When a request to begin an ADS-B recording is received, a function is called which begins two new threads. The first is used to record the ADS-B messages, while the second is used to monitor the first thread and end it if either the timeout is reached or a KILLRCRD command is received.

The receiver software was configured to distribute messages passed to it from the ADS-B receiver over the ethernet interface, so that the recording thread can collect them from the local loopback ethernet interface. The receiver software can distribute these messages using either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP). Using TCP offered an improved timestamp resolution compared to UDP, but the first version of the flight application script used UDP due to an issue where multiple socket connection timeouts would cause the receiver software to crash. This issue was eventually solved after launch, though, and the application script was updated about half-way through the ADS-B data collection campaign, in early 2017.

6.6.4 On-Orbit Software Updates

Software Requirement 9 states that the application script must be re-writable on orbit. The original plan for the ADS-B payload was to use the same version of the application software throughout its six month mission, but a bug fix during the ADS-B data collection campaign showed the promise of improving the message timing precision by using TCP rather than UDP to collect data messages over the local loopback network interface.

It was decided that the software would be updated after the primary and secondary objectives of collecting ADS-B data over the North Atlantic Ocean and Australia were accomplished. This was in case there was an issue during the software upgrade which caused the payload to become non-responsive. The process had been tested successfully on the ground several times, but as the improved message timestamp accuracy was a nice-to-have rather than a requirement, any unnecessary risk that could impact the mission’s success was avoided. The software update took place in early 2017, and no issues appeared during the process. Regular operations were able to resume during the following orbit with the updated application script.
Figure 6.4: Flowchart describing the ADS-B message recording process using TCP

6.7 Planestation Ground Software

In addition to the software used to control the payload, software was written to store and display the ADS-B data once it was downlinked to the ground. This software is called Planestation, and consists of two separate but related software tools written to support the ADS-B mission. The first is a data file
parser and database creation tool, and the second is a plotting utility for displaying aircraft detected by the ADS-B receiver.

### 6.7.1 ADS-B Database

Once an ADS-B data file is transferred from the payload to the spacecraft OBC, the file is compressed to reduce its size for downlink. The compressed data files are stored on the spacecraft OBC until a ground pass, at which point they are transferred to the ground to be processed. Once downloaded, the files are decompressed and stored in a “Raw Data” folder.

The raw data files are compact, which makes them useful for data downlink and storage, but they are not easy to search and extract useful data from. Once it is downloaded, the data must be parsed into a format that is more organized and easily searchable so that it can be used for analysis. To do this, an SQLite3 database is used to store the ADS-B data. SQLite3 is freely available, simple to use, well integrated with Python, and there are many software tools available to view and search databases written in the language. Figure 6.5 shows the ADS-B database open in SQLite Manager, a free SQLite3 database browser distributed as a Mozilla Firefox extension.

![Figure 6.5: The CanX-7 ADS-B database, viewed in SQLite Manager](image)

The database is intended to make it easy to access aircraft data and generate plots of aircraft, as well as to do simple data analysis. Data in ADS-B messages, as described in Chapter 2, is encoded using the fewest number of bits possible to keep messages compact. Most high-level programming languages, however, use the Byte as the basic unit of storage, and as such are designed to store, read, and process data in Bytes rather than bits. This means that in the database it is better to store the ADS-B data in more typical formats, such as integers, floating point numbers, and character strings. Even though this increases the amount of storage the database requires, it simplifies the construction of search queries and reduces the time it takes searches to be performed. The increase in storage required for the database compared to the raw data files is not a large concern, as storage on the ground is far less constrained.
than while on orbit.

The database as constructed contains fields for key data included in the three main types of ADS-B messages the CanX-7 mission is interested in collecting: airborne position, airborne velocity, and flight information messages. While other types of messages are stored in the raw data files, they are omitted from the database. Each database entry is tagged with a unique identifying number for the message it represents, a recording number unique to each file entered into the database, and the raw data file name. Also included are the message time of reception, the multilateration timer value and signal strength values attached to each message by the receiver software, and the raw message in binary format. Raw messages are parsed by an automated parsing tool that extracts the data from the message, converts it to the proper format, and stores it in the database. The transmitting aircraft ICAO code and message type are available in every message received, while fields for specific message types such as aircraft call sign, aircraft heading and velocity, altitude, and latitude and longitude are also included. For position messages, two of which must be received before a position can be calculated, the even/odd status of each message and the Compact Position Reporting (CPR) latitude and longitude of each message is also present.

### 6.7.2 ADS-B Data Parsing Tool

The purpose of the ADS-B data parsing tool is to automatically extract information from the ADS-B data files and store it in a database which can be viewed and searched, and from which information can be extracted. PlaneStation Parser is a Python script that runs in a terminal window without a Graphical User Interface (GUI) and manages the data files received from the spacecraft and the ADS-B database. The flowchart in Figure 6.6 gives an overview of how the parser works.

The ADS-B parser begins by initializing the database if it can be found, and creating a new SQLite3 database if not. The parser then goes into a loop, where it checks every five seconds for the presence of raw payload data files in the folder where they are stored after being received from the satellite. If one or more data files are found in the folder, the script enters its parsing loop. Once in the parsing loop, the script opens the first file and searches it for the beginning of an ADS-B long message, the hex sequence **0x1a 0x33**. During operations, Mode A/C messages were not recorded, and short ADS-B messages were not stored in the database.

Once a message is found, the signal strength is stored in the database and the MLAT timestamp is converted to an absolute time based on the file creation time and stored. The ADS-B frame is then broken into binary data bits. Data is then extracted, parsed based on the message format, and stored in the database. For call sign and velocity messages, the data can be directly extracted. Due to the Compact Position Recording (CPR) format used for position messages, two messages are required to determine an aircraft position, so it is a more complicated process to parse these messages.

When a position message is detected by the parser, the intermediary CPR values are calculated and stored in the database along with the other information contained in the message that be extracted from the single frame. More information on decoding ADS-B position messages can be found in [42]. When transmitted by an aircraft, position messages are labelled as either even or odd, denoted in the database as 0 or 1, respectively. When a position message is found, an entry is added to a Python dictionary that stores the most recent even and odd CPR latitude and longitude values for each aircraft. If both even and odd CPR latitude and longitude values are in the dictionary, the aircraft’s actual latitude and longitude are then calculated and stored in the database as well.
6.7.3 Aircraft Plotting Tool

A simple aircraft plotting tool with a graphical user interface was created in order to simplify the process of searching the database and generating plots of aircraft flight paths. The interface for this software tool is shown in Figure 6.7.

The plotting tool gives the user several options for restricting the aircraft to be plotted. Individual aircraft or flight routes to be plotted by filtering by ICAO address or by callsign. Aircraft to be plotted can also be restricted by a specific range of times or recording number. In addition, all messages in the database can be plotted unfiltered. The plotter works by constructing SQLite3 queries using Python, and then executing them on the database. Messages that match the query then have their ICAO identifiers, entry numbers, recording numbers, latitudes, longitudes, and altitudes stored in a .csv file. This .csv file is then read back by the script, and all the latitude and longitude pairs are plotted over a NASA
composite satellite image of the Earth. Figure 6.8 shows a plot of all of the position messages received by the payload over the seven month ADS-B recording campaign. A 90% transparency was applied to each plotted point to reduce overcrowding of the plot.
6.8 Conclusion

Software was written for the CanX-7 payload, both for command and control of the payload and for data management during operations. Several scripts were based on pre-existing prototype software, but much of the architecture was redesigned for greater functionality, flexibility, and reliability. Requirements were generated, with the final version of the flight software compliant with most.

<table>
<thead>
<tr>
<th>Number</th>
<th>Requirement</th>
<th>Compliant?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Communication via NSP</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>2</td>
<td>Implementation of standard NSP commands</td>
<td>PARTIALLY COMPLIANT</td>
</tr>
<tr>
<td>3</td>
<td>Payload computer responsiveness</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>4</td>
<td>Boot from microSD</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>5</td>
<td>Use Angstrom Linux OS</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>6</td>
<td>Dual storage partitions</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>7</td>
<td>Software to partition distribution</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>8</td>
<td>Bootable if writeable partition is corrupted or unavailable</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>9</td>
<td>Software updatability</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>10</td>
<td>Use of standard FPGA and receiver software interface</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>11</td>
<td>Payload internal clock</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>12</td>
<td>Payload computer logging</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>13</td>
<td>Linux terminal interface</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>14</td>
<td>Payload computer underclocking</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>15</td>
<td>ADS-B message timetagging</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>16</td>
<td>Automatic file transfer</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>17</td>
<td>Configurable recording parameters</td>
<td>COMPLIANT</td>
</tr>
<tr>
<td>18</td>
<td>Telemetry collection</td>
<td>COMPLIANT</td>
</tr>
</tbody>
</table>

The only software requirement that was not fully met was the requirement to implement the standard NSP commands in the control software, with some being incompatible with the operating system of the payload computer. The commands that were not implemented, though, had their functionality replaced through implementation of similar commands.
Chapter 7

Payload Electronics Testing

The CanX-7 ADS-B receiver payload hardware is an updated version of the hardware originally selected and developed by the Royal Military College of Canada for the prototype payload [25]. When flying any piece of hardware on a satellite, particular one with commercial off the shelf (COTS) components not designed for space applications, it is important to rigorously test it, in order to ensure it will not fail once in space. With the payload control software complete, several tests were designed to categorize the performance of the receiver and its control software.

Testing was broken down into three categories:

1. Performance categorization
2. Unit-level testing
3. System-level testing

For each test to be run, a test plan was developed, with requirements that had to be met before the receiver could be considered flight-ready. Three receivers were purchased at the beginning of the testing process, one designated as the flight model, one as a flight spare, and one as an engineering model. If the flight unit had failed any of the tests and the resulting failure was traced back to a defect in the receiver build quality, it would have been replaced with the spare unit, and the tests would have to have been run again on the new unit.

7.1 Testing Software and Setup

The CanX-7 ADS-B payload is the first receiver of this type to be flown on an SFL spacecraft. As explained in Chapter 2, ADS-B uses an amplitude modulation scheme known as Pulse-Position Modulation (PPM) to encode data onto its carrier signal. This type of modulation is uncommon in spacecraft systems, and as such, testing equipment was not readily available at SFL and had to be developed specifically for this project. The emphasis for the testing setup was placed on rapid software development, along with ease of access to and low cost of the necessary hardware.

The basic goal of most of the tests was to ensure that the receiver would properly decode ADS-B messages under realistic operating conditions, and to measure its level of sensitivity. To do this, during each test data was transmitted to the receiver at decreasing input power levels under various operating
conditions, and recording the percentage of messages that were decoded successfully to determine the sensitivity of the receiver. As the portion of the receiver software that handles decoding messages is proprietary to the manufacturer, it was necessary to be able to transmit full ADS-B messages, with proper formatting, to be detected by the receiver and passed off to the payload application script. From here, the data can then be collected and analyzed to determine the receiver sensitivity. A flow chart showing the basic steps of a sensitivity test is shown in Figure 7.1.

Figure 7.1: CanX-7 ADS-B payload sensitivity test

7.1.1 Software-Defined Radio

During software development, a roof-mounted ADS-B antenna with integrated pre-amplifier was used to verify that the software was correctly recording ADS-B message data. During hardware acceptance testing, however, the messages received by the receiver had to be more tightly controlled, so the relative numbers of received and missed messages could be used to determine packet-error rates, and correlate them to a known input signal strength.

To transmit ADS-B messages to the payload, a software-defined radio (SDR) was selected over a traditional signal generator and modulation source. A software defined radio is a radio where the components of the signal chain have been implemented in software rather than hardware. The two advantages this solution has over a signal generator are cost and ease of implementation. A traditional signal generator can cost upwards of $10,000 CAD, but a high-quality software defined radio can be purchased for around a tenth of the price. In addition, many SDRs also contain an on-board Field-Programmable Gate Array (FPGA), programmable via USB and Windows or Linux-based software such as Matlab or GNURadio, which means that integration with a new test setup is quick and easy.

The disadvantage that an SDR transmitter has over a more traditional signal generator is that the range of power levels at which the SDR can accurately produce a noise-free signal is more limited. This is not an inherent quality of an SDR, it is rather that most inexpensive SDRs are geared towards receiving and demodulating signals or transmitting at relatively high power levels. They are not designed for high-precision output at the levels that the sensitivity test requires. To test the sensitivity of the ADS-B
receiver, the signal at the input to its LNA needs to be quite low, in the range of -80 to -100 dBm. To work around this limitation, a high precision adjustable attenuator was used to reduce the transmitted signal level from that output by the SDR down to the range needed during sensitivity testing.

The USRP B100 Software-Defined Radio

The SDR selected for payload electronics testing is the Universal Software Radio Peripheral (USRP) B100, from Ettus Research. The USRP B100 has two channels, one dual purpose transmit and receive channel, and one receive only channel. It is self contained within an enclosure, powered via a 6 Volt DC input, and accepts data via a high-speed USB 2.0 port. The B100 is compatible with MATLAB, LabView, and GNU Radio programming and control environments through the use of the USRP Hardware Driver (UHD).

![USRP B100](image)

Figure 7.2: The Ettus USRP B100 software-defined radio

Figure 7.3 shows a simplified block diagram of the USRP B100. A Xilinx Spartan 3A FPGA handles data input and output via the USB 2.0 port. When transmitting, the FPGA uses an interpolating Cascaded Integrator-Comb (CIC) filter to generate a modulated digital signal at baseband. This is then passed through a digital upconverter and a dual 128 Megasample per second digital to analog converter. The USRP then passes the signal to a daughterboard, which handles frequency upconversion, amplification, and transmit/receive channel switching.

GNU Radio and Controlling the SDR

The USRP B100 is compatible with several control environments, including GNU Radio, MATLAB, and National Instruments LabView. GNU Radio was selected over MATLAB and Labview because of its ease to use interface and the author’s familiarity with using it to develop USRP applications. GNU Radio is an open-source software development kit that implements various types of signal processing functions as blocks for use in software-defined radio applications. It can be used with a variety of SDR hardware peripherals, or on its own as a simulation tool. It is fully compatible with the USRP line of products through the use of the Ettus USRP Hardware Driver (UHD) for both transmitting and receiving.

To control the USRP and provide it with a transmittable data stream, GNU Radio was installed on a dedicated laptop running Ubuntu Linux. A graphical user interface, called GNU Radio Companion,
Chapter 7. Payload Electronics Testing

Chapter 7. Payload Electronics Testing

Figure 7.3: A simplified block diagram of an Ettus USRP B100 [3]

is bundled along with the installation of the GNU Radio software, and was used in order to generate the control blocks to manage the USRP. Figure 7.4 shows the blocks which are used to generate the simulated ADS-B messages transmitted via the USRP.

Figure 7.4: GNU Radio Companion definition file for ADS-B sensitivity testing

A binary file containing a chip sequence representing the baseband ADS-B messages is streamed to the USRP at 2 Msps. The file is first unpacked, converting each byte into 8 chips, each of which is a 1 or 0 representing a pulse or null. These chips are each stored as an integer. The integers are each then converted to floating point numbers, multiplied by 0.4 to prevent clipping, and then converted to a complex I&Q value, which is then streamed to the USRP and modulated onto the carrier frequency of 1090 MHz. Note that the file source is only streamed into the real section of the complex number conversion block, the imaginary input is left empty. This means that the generated signal will only have an in-phase component. Details about the file containing ADS-B information for the testing can be found in Section 7.1.2.

Note that there are several grayed out blocks in Figure 7.4, indicating that they are disabled. The
vector source block contains a simplified sequence of the chips stored in the ADS-B binary file, only containing a single message and its preamble. This was used as a modulation source during initial testing while experimenting with the USRP and the receiver, and during the power calibration step of the electromagnetic-compatibility (EMC) test described in Section 7.4.1. The signal source block is used to generate a simple carrier wave at 250 kHz above the ADS-B frequency, which was initially used to test and calibrate the output power of the transmitter (see Section 7.1.3). The WX GUI Scope Sink block was used to view the output spectrum of the modulated signal in baseband.

### 7.1.2 ADS-B Message Generation

The basic structure of an ADS-B message is presented in Section 2.2. It consists of a preamble to time-synchronize the receiver, a message type identifier, aircraft status indicator, aircraft identification address, a data message, and a Cyclic Redundancy Check (CRC) code. In order for the receiver to decode and pass off to the application script any messages it receives, all of these components must be present in the demodulated packets. As described in Section 2.2.2, pulse-position modulation is an amplitude modulation format used to transmit data as part of the ADS-B standard. Each time slot is one microsecond long and is divided in half, with a pulse in the first half representing a binary 1, and a pulse in the second half representing a binary 0.

For clarity in this section, PPM pulses will be described as PPM chips, whereas data will be described as data bits. In this way, the PPM chip sequence “01” represents the data bit “0”, whereas the PPM chip sequence “10” represents the data bit “1”. To drive the USRP, modulation data needed to be streamed to it over the USB port at a very high rate. To do this, a static file was generated prior to the test containing all of the messages to be transmitted, rather than trying to generate messages in real time. This approach reduced the amount of work required of the Linux host computer, so that all it had to do was stream data read from a file.

### MATLAB ADS-B File Generator

A MATLAB script was created to generate the test file. This allowed multiple files to be generated quickly, each containing different data if desired. Each generated file contains 1000 ADS-B flight information messages, each tagged with a unique identification number, with a dead zone between each message to allow the receiver time to process and store it. While it is realistic to assume that multiple messages will arrive within a short time frame and interfere with each other during on-orbit operations, this specific case is not what is being examined during sensitivity testing. Rather, it is the sensitivity limit of the RF components of the receiver. As such, the break between each message sent ensures that messages will not be dropped due to a packet collision or lack of processing resources on the payload computer.

The data message consists of six parts: the preamble, the message type identifier, the flight status indicator, the aircraft identifier, the message data, and the cyclic redundancy check (CRC) code. Other than the preamble, the entire message was generated as binary data first, and then converted to PPM chips at the end of the file generation, in order to make use of some of the bit manipulation functions built into Matlab. The preamble was generated directly as binary 1 and 0 chips, as it contains sections without any pulses that don’t map properly to the PPM format.

All of the ADS-B messages generated in each of the files were of the aircraft identification (flight
information) type. This message is simple to encode, containing only an eight character string of letters and numbers normally used to identify an aircraft's airline and flight number. The ADS-B data generation script assigns each message a unique identification number and stores it in this data field rather in place of aircraft information. The first two digits of the identification number represent the file ID number, the third digit is left blank, and the last four digits represent the message number within the file. Since each message has a unique identifier, it is possible to determine whether any messages that are not detected are missed in a way that is expected (e.g. due to normally distributed bit errors at low input power levels) or unexpected (e.g. clustered due to an intermittent amplifier power loss, cable disconnection, or external noise source).

7.1.3 USRP Transmitter Performance Categorization

When performing any radio receiver testing, it is important to ensure that the performance of the transmitter is calibrated so that the input level at the receiver is correct. Since a pre-calibrated signal generator was not used as the RF source during testing, it was necessary to measure the output power of the USRP using a well-calibrated spectrum analyzer.

**USRP Transmitter Power Calibration**

To calibrate the transmitted power of the USRP, the output port was connected through an adjustable attenuator to an Agilent N9030A PXA signal analyzer. The USRP was configured to transmit by activating the Signal Source block chain from Figure 7.4, and disabling the File Source block chain. The signal source block generates a sine wave at a specified frequency, in this case 250 kHz. The sine wave amplitude is then decreased to 40% of its maximum value to prevent clipping, and modulated onto the carrier frequency. This causes the USRP to transmit a carrier wave at 250 kHz above the carrier frequency of 1090 MHz. Since all of the power of the signal is contained at this carrier frequency, the transmitter power is calculated by measuring the peak level of the signal. After measuring the power of the transmitter, the attenuation was increased in steps of 5 dB from 60 dB to 80 dB, and then in steps of 1 dB from 80 dB to 85 dB, to ensure that the curve was linear. The results of the power calibration are shown in Figure 7.5. The power decreases linearly with increased attenuation, as expected. With 0 dB of attenuation, the power output of the USRP plus the losses associated with the coaxial cables used to connect it to the unit under test was calculated to be -11.7 dBm. This setup was used during the RSSI calibration, thermal shock, and thermal acceptance tests.

7.2 Payload RSSI Calibration

Originally, it was believed that the receiver had an on-board method of received signal strength measurement that could be used to determine the input power of any ADS-B signals that were received, useful for verifying performance on orbit. With the USRP output power calibrated, the received signal strength indicator (RSSI) on the board could then be calibrated. For this calibration, the same set up was used as during the power calibration, but the spectrum analyzer was replaced with the flight payload hardware, including the LNA.

To conduct the test, 1000 messages were transmitted from the USRP and recorded by the payload for a number of attenuation values beginning at -60 dB, and ending at -85 dB, in the same steps as
Chapter 7. Payload Electronics Testing

Figure 7.5: USRP measured power compared with the attenuation value

were used during the power calibration. This corresponds to an input power level at the LNA from -71.7 to -96.1 dBm, greater than the bounds that the receiver is expected to see on orbit and be able to decode successfully. Note that these values are calculated using the calibrated power measurements, not by adding the attenuation used to the input power level at 0 dB attenuation, so they differ slightly. A Python script was written to process the recorded data files, extract the signal strength measurement bytes from each recorded frame, and plot the values. Four plots are shown in Figure 7.6 of the RSSI measured by the receiver at different input signal levels.

Each of the messages transmitted to the receiver by the USRP was sent at the same transmit power level. However, it can be seen from the plots of the RSSI byte values that the measurement is unstable, with a 1σ standard deviation of 10% or more in each of the four measurements. To determine whether the average RSSI value could be used as a reliable signal strength measurement, the test was re-run several times at each power level. When more than approximately 40 messages were received during a test, the average value of their RSSI bytes did appear to be consistent and correlatable to signal input power.

Unfortunately, while on orbit it was not expected that enough messages would be received in a short enough period of time to perform this correlation. As explained in Section 2.2.1, an aircraft will transmit messages only a few times per second, at most, and these messages are split between top and bottom mounted antennas, with the payload only expected to receive messages from the top-mounted antennas. At the speed the satellite moves, an aircraft will not be located in one spot relative to the spacecraft for a long enough time to transmit enough message for a reliable correlation to be formed to predict an exact value of the signal power. Even if this indicator was unable to be used as a precision indication of received signal strength, it was thought that a rough idea could be formed of how powerful the signal was from a particular aircraft.

Due to the high standard deviation in the measurements, further investigation was done to determine the cause of the instability, in an effort to determine if anything could be done to decrease it. After conversations with the manufacturer of the receiver, it was found that what was believed to be a RSSI measurement was actually more akin to a signal-to-noise ratio measurement. As the receiver
was not originally designed for use in a spacecraft, the signal strength byte recorded with each message is measured with reference to the noise floor rather than a DC ground. To obtain the signal strength, the receiver samples the message preamble at 16 MHz. Each pulse in an ADS-B message preamble is 0.5 µs long, of which there are four in the preamble over the course of 8 µs. The average of the 96 noise measurements are subtracted from the average of the 32 pulse measurements to obtain the signal strength value. Figure 7.7 shows a simulation of how this method can cause two signals, one with a low noise-floor (top) and one with a high noise floor (bottom), to appear to have different levels even though the actual signal strength is the same. The actual signal level is the blue line, while the red X’s represent sampling locations. In both plots, the signal pulses have an average signal strength of 1, but in the bottom plot, the noise floor is 4 times higher than the top plot. The resulting signal strength measurement using the receiver’s method would be 0.9 for the top plot and 0.6 for the bottom plot, even though the actual signals have the same power.

During testing, it appeared that the noise was referenced to ground, but this only appeared to be the case because the noise level was consistently low. While on orbit, the noise level was expected to vary depending on several factors, and could not be measured easily to calibrate the levels against. Because of this, the signal strength byte is not expected to be correlatable in even an approximate way to the power of the received signal.
7.3 Unit-Level Testing

Environmental testing of hardware is a key component of the microspace design philosophy. Each piece of hardware, before being accepted for use on a spacecraft, must be tested over temperature to make sure that it can withstand the extremes of the space environment without damage or significant performance degradation. At SFL, all hardware must undergo a thermal shock test to ensure workmanship quality, and a thermal acceptance test, where the performance of the unit is verified at previously specified temperature limits.

7.3.1 Functional Tests

A functional test is one whose purpose is to verify the functionality and performance of a hardware unit. At SFL these tests are typically broken down into two types: a long-form functional test (LFFT) and a short-form functional test (SFFT). A LFFT typically will test all manner of functions on the unit including power consumption, internal current draw and voltage regulation, and proper response to and execution of commands. Hardware may also go through more additional tasks specific to their purpose. For example, additional tests for radio frequency receivers include sensitivity and RSSI tests, and computers undergo tests to confirm that they can load and run software from memory, and that their storage can be read from and written to. A SFFT will typically consist of a smaller subset of tests that can be run in less time, and is used to confirm basic functionality of the unit as opposed to categorizing performance.
The ADS-B receiver must be attached to the payload computer for it to operate, as the payload computer must flash an FPGA image to and communicate constantly with the receiver for it to function properly. This interconnectedness meant that a single LFFT and SFFT had to be created for the payload as a whole, rather than breaking the components up and testing them individually. Several tests were performed as part of each LFFT, with each test run at two different bus voltages, 4.2 V and 4.8 V. As part of the 4.2 V test, the payload was booted at 4.8 V, then transitioned to 4.2 V after ensuring it was operational. The procedure for the LFFT was as follows:

1. Check responsiveness to ping commands
2. Transition from pseudo-bootloader to application script
3. Verify payload software state when idle and during recording
4. Check temperature of payload
5. Verify that ADS-B file size is displayed correctly
6. Ensure that ADS-B messages are received from the USRP transmitter at a high power level
7. Perform full sensitivity sweep on payload receiver
8. Test data file transfer to OBC and check file integrity
9. Test shutdown sequence, ensure payload current draw is reduced to expected levels when off

The SFFT is intended to be a simplified version of the LFFT, but must still be able to confirm that the receiver is functioning properly. As such, the procedure for the SFFT was as follows:

1. Check responsiveness to ping commands at boot voltage (4.8 V) and operational voltage (4.2 V)
2. Verify payload software state at operational voltage (4.2 V)
3. Check temperature of payload at operational voltage (4.2 V)
4. Test shutdown sequence at operational voltage (4.2 V), ensure payload current draw is reduced to expected level when off

The reason for performing tests at both 4.2 V and 4.8 V is that the nominal operating bus voltage for the spacecraft is only regulated to 4.2 Volts if the bus voltage falls below that level, at which point the spacecraft Battery Charge Discharge Regulator (BCDR) will increase it by drawing power from the spacecraft battery. If the voltage is higher than this setpoint, however, the BCDR will not reduce it. Effectively, the spacecraft bus voltage is allowed to float at or above 4.2 Volts. When in sunlight, this actually will happen often, with the bus voltage floating anywhere between 4.2 V and 5.3 V. As the payload was designed to operate in both sunlight and eclipse, it was important to quantify any performance differences arising from changes in the bus voltage. For this reason, both thermal sensitivity and EMC testing were performed at different bus voltage levels to ensure on-orbit performance.
7.3.2 Thermal Shock Testing

The CanX-7 ADS-B payload has three printed circuit boards that must be verified for workmanship quality: the external LNA, the receiver, and the payload computer. After the modifications described in Section 2.3.2 were performed and a visual inspection of all the components on each of the boards was performed by a technician, the boards were tested to ensure proper functionality. A LFFT was performed on the combination of the LNA, receiver, and payload computer in a flight-like configuration, after which the boards were sent to an external facility for thermal shock testing.

As part of a thermal shock test, the board is rapidly heated and cooled to its operational temperature limits, at a rate of 25°C per minute. The goal of the test is to rapidly heat and cool the solder joints on the boards, stressing the metal to the point of cracking the solder joints if they contain any flaws. After the thermal shock test, each board is re-inspected by a technician. The CanX-7 ADS-B receiver flight boards showed no damage due to the test, so another LFFT was performed to verify functionality. After all payload components passed all tests, they were moved forward to unit-level thermal acceptance testing.

7.3.3 Thermal Acceptance Testing

After the functionality of the receiver was confirmed and the workmanship quality of the printed circuit boards was verified via thermal shock testing, the boards were tested for functionality at temperatures representative of the space environment.

Maintaining acceptable internal temperature ranges is an important part of spacecraft design. In order to properly design the thermal control system, the operational and survival temperature ranges of each of the subsystems must be known. Thermal simulation of the spacecraft in its operational modes is then used to ensure that all of the subsystems maintain proper temperatures in both worst-case hot and worst-case cold conditions. When designing a thermal control subsystem, the larger the operating and survival temperature ranges of each of the other subsystems, the better. The ADS-B receiver on CanX-7 was tested and accepted over a range of -20°C to 60°C operating temperature, with survival minimum and maximum temperatures of -30°C and 70°C, respectively.

Payload Receiver Thermal Test Setup

In preparation for the thermal acceptance test, a set of three thermocouples was attached to the payload using RTV, a clear, flexible staking compound resilient to extreme temperatures, and Kapton tape. One sensor was placed on each part of the payload: one on the casing of the LNA, one on top of the payload computer’s processor, and one below the FPGA on the receiver. During the course of the test, these temperatures were all individually monitored and recorded.

With each of the thermocouples attached to the payload, it was placed in an anti-static bag to protect the components from unintended electro-static discharges, which can damage components on the payload and lead to poor performance or even complete failure. Before sealing the anti-static bag, desiccant was placed inside it beside the payload boards. The desiccant serves to remove any moisture from the air during the test. As the ambient air temperature approaches the freezing point of water during the test, any moisture in the air will condense and freeze on the inside of the anti-static bag and on the payload boards. When the temperature is then raised above the freezing point of water, the ice will melt onto
the payload. Any liquid water on the boards can cause shorts, damage to their components, or even damage to the underlying PCB itself, any of which would render the unit unsuitable for flight.

To conduct the test, the payload was placed in one of SFL’s thermal chambers. Power and ground lines, UART control lines, and a coaxial cable input were fed into the chamber via an opening on the side of the chamber. Power for the unit was supplied by a DC power supply, while a laptop was used to control the payload, monitor board temperatures, and manage the thermal chamber. A second laptop running Linux was used to control the USRP ADS-B transmitter, whose RF output was connected to an adjustable attenuator connected to the payload LNA RF input. Temperature minimum and maximum safety limits on the chamber were set to -40°C and 85°C, respectively, and the dry air feed system was enabled. The dry air system feeds air from which the moisture has been removed into the thermal chamber during the test, as opposed to air from the ambient environment. This serves as an additional layer of protection against ice crystal formation during the test.

**Payload Receiver Thermal Test Profile**

The thermal acceptance test performed on the ADS-B receiver payload consisted of seven sections, each verifying a different aspect of the receiver’s performance across temperature. The test temperature profile is shown in Figure 7.9. It is important to note that in this test, all temperatures in the test profile were treated as “interface temperatures”, or those of the ambient environment. The component temperatures were allowed to get hotter or cooler than this during the course of the test. All temperature slews during the test were performed as close to 2° per minute as possible. Slower slew rates were allowed, in particular when approaching a target temperature, but the slew rate was never allowed to be higher.
The test began and ended with LFFTs performed at room temperature. The first test acts as a baseline for the unit performance during the test, while the final test serves to highlight any permanent shift in the performance of the unit due to the thermal test. After the initial room temperature test, the unit was turned off, the temperature of the thermal chamber was raised to 70° Celsius (C), and the payload was allowed to soak at this temperature for one hour. The temperature was then reduced to the payload’s maximum operating temperature of 60° C, at which point another LFFT was run to characterize the performance. The payload was then turned off, and the temperature reduced to -30°, the cold survival limit of the unit, where it was again allowed to soak for one hour. The temperature was then raised to its minimum operating temperature of -20° C, and another LFFT was run. The hot and cold LFFTs are performed to ensure that the payload still functions after being exposed to its storage temperature extremes, and also to categorize the payload’s performance at both hot and cold temperatures.

After the hot and cold tests, the payload was run through two temperature cycles without being power cycled. To ensure the payload continued functioning as expected, an SFFT was run at each of the cycle plateaus (hot and cold) before the payload was returned to room temperature for the final LFFT. The thermal cycles ensure that if the payload does see large temperature swings while operating, for example a sunlight to eclipse transition, that nothing unexpected affects its performance or causes it to fail. During the thermal acceptance test, the unit passed all SFFTs and LFFTs.

**Thermal Sensitivity Testing**

A sensitivity test was performed as part of each LFFT in order to measure the message decoding percentage of the payload at each of its operational temperature extremes. Ideally, the payload would suffer little performance degradation at both temperature extremes and when operating at both 4.2 V and 4.8 V. During each test, 1000 ADS-B messages were transmitted from the USRP to the receiver,
with the decoding percentage calculated by counting the number of received frames correctly decoded in the recording files.

During the test, the ADS-B receiver was able to maintain its decoding performance at both hot and cold temperatures in most cases. Figure 7.10 shows the number of successfully decoded messages under various operating conditions. When operating at 4.8 Volts, the performance of the payload is consistent across all temperature ranges. When operating at 4.2 Volts, the performance is similarly consistent except at -20° C near the sensitivity limit of the receiver. Even though the performance degradation was minor (<1dB sensitivity change), it was noticeable compared to operation at room temperature and +60 °C, and so consideration was given to changing to the spare unit.

Ultimately, the flight payload was successfully verified over temperature, and continued to integration and system level testing. Based on the thermal analysis performed as part of [17], there were not expected to be many if any cases of the payload having to operate at this temperature range. Thermal analysis of the spacecraft showed that while the payload could reach as low as -18.4° C under worst-case cold
conditions, these would only happen with the spacecraft in safe hold mode, where most of the spacecraft subsystems including the transmitter, attitude control system, OBC, and ADS-B payload are turned off for the entirety of the orbit [17]. As this unit is a payload receiver and not a telemetry and command receiver, it does not need to operate in this spacecraft mode. Nominal temperatures expected for worst-case cold and worst-case hot orbits in a payload operational mode were between -7\(^\circ\)C in a cold reference orbit and 14\(^\circ\)C in a hot reference orbit [17].

7.4 System-Level Testing

In addition to testing at the unit level, it was also important to test the receiver while integrated into the CanX-7 spacecraft to gain a better understanding of the performance and identify any issues that may arise from being integrated with the other avionics. System-level testing of the payload consists of functional, electro-magnetic compatibility and thermal vacuum testing. Functional testing is simple, and essentially consists of a unit-level LFFT while integrated and operating off of the power provided by the spacecraft bus. Once the payload is integrated into the spacecraft, the coaxial input port of the LNA is connected to the antenna, so it is impossible to feed USRP transmitted messages directly into the payload. This portion of the test is instead replaced by recording messages directly from passing aircraft. The spacecraft was placed on the clean-room bench with the antenna directed upwards. Messages were recorded for a fixed time interval, and the resulting file was run through ADS-B database parsing software to verify that it contained properly formatted ADS-B messages. A sensitivity test was not performed as part of this test, as it would be difficult to isolate the spacecraft from external interference, which could skew the results significantly. Since the sensitivity would be measured during electromagnetic compatibility testing and was unlikely to change once integrated, there was little risk in skipping this component of the test.

7.4.1 Electromagnetic Compatibility Testing

During electromagnetic compatibility (EMC) testing, the sensitivity of a receiver is measured after being integrated with the rest of the spacecraft. Each receiver on the spacecraft is tested several times, once for each of the modes in which it must operate, with the other avionics in an operational state representative of the spacecraft mode being tested. For a command receiver, it is important to verify that it works properly in all spacecraft modes, so the spacecraft cannot get stuck in a situation where it is impossible to command. Since the ADS-B receiver is a payload receiver and its functionality is not critical to the survival of the spacecraft, it is only tested in two operational modes, nominal operations (CANOE) and nominal operations, transmitting (CANOE, TX). Table 7.1 shows which spacecraft subsystems are powered on or off during the two testing modes. In each of the two modes, sensitivity measurements were also taken at several different bus voltages.

EMC Test Setup and Calibration

To perform the EMC testing, the spacecraft was set up in an anechoic chamber in the SFL clean room along with the USRP transmitter and the radio ground support equipment (GSE) rack, as in Figure 7.11. Two laptops were used to control the test, a spacecraft control laptop connected to the radio GSE rack, and a USRP control laptop connected to the USRP transmitter. The radio GSE rack was used to
Table 7.1: ADS-B payload EMC testing configurations

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>CANOE</th>
<th>CANOE, TX</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>OBC</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>Magnetorquers</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>Magnetometer</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>Minicam</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>Transmitter</td>
<td>OFF</td>
<td>ON</td>
</tr>
</tbody>
</table>

communicate with the spacecraft and control it wirelessly between measurements, as well as to download measurement data.

As the objective of the EMC test is to observe any performance degradation in the sensitivity of the receiver, it is critical to make sure that the input power to the receiver is calibrated correctly when running the tests, or else the results will be incorrect. To calibrate the input power to the payload receiver, the test was set up similarly to the diagram in Figure 7.11. In lieu of the spacecraft, the tinsat used during antenna pattern testing was set up with an EM ADS-B antenna and fed into a pre-amplifier,
and then through a 15-foot test cable and into a spectrum analyzer.

Data was transmitted via the USRP using both Signal Source and Vector Source block chains shown in Figure 7.4, with the Vector Source set to repeat the message after transmitting. The channel power was recorded in the 2 MHz span centered on 1090 MHz using the spectrum analyzer at attenuation levels from 0 to 20 dB. Power numbers were also measured at higher values of attenuation, but at this point that the power of the modulated signal began to approach the noise floor of the spectrum analyzer, and so the measured values started to become nonlinear with increasing attenuation. For values of attenuation used during the test that were higher than those measured in Table 7.2, a best fit linear regression for the data from 0 to 20 dB was used to model the input power level.

<table>
<thead>
<tr>
<th>Attenuation Value (dB)</th>
<th>Measured Power (dBm)</th>
<th>Receiver Input Power (dBm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-39.19</td>
<td>-52.05</td>
</tr>
<tr>
<td>2</td>
<td>-41.24</td>
<td>-54.10</td>
</tr>
<tr>
<td>4</td>
<td>-43.24</td>
<td>-56.10</td>
</tr>
<tr>
<td>6</td>
<td>-45.38</td>
<td>-58.24</td>
</tr>
<tr>
<td>8</td>
<td>-47.34</td>
<td>-60.20</td>
</tr>
<tr>
<td>10</td>
<td>-49.19</td>
<td>-62.05</td>
</tr>
<tr>
<td>12</td>
<td>-50.77</td>
<td>-63.63</td>
</tr>
<tr>
<td>14</td>
<td>-52.59</td>
<td>-65.45</td>
</tr>
<tr>
<td>16</td>
<td>-54.77</td>
<td>-67.63</td>
</tr>
<tr>
<td>18</td>
<td>-56.61</td>
<td>-69.47</td>
</tr>
<tr>
<td>20</td>
<td>-58.62</td>
<td>-71.48</td>
</tr>
</tbody>
</table>

EMC Test Results

The results of the EMC test are shown in Figure 7.12. With the spacecraft at its nominal bus voltage, it was found that the performance of the receiver was only slightly degraded from that measured during the thermal acceptance testing. The dashed lines in the figure show the best and worst case expected input power levels of messages received from aircraft in the hemisphere of the antenna. The worst case is for an aircraft located at the horizon, while the best case shown is with the optimal combination free-space loss, gain from the spacecraft antenna, and gain from the aircraft antenna. A polarization loss of 3 dB was assumed in both cases.

Performance during this test was similar to that seen during thermal acceptance testing, but with slight desensitizing when the spacecraft bus voltage was set to 4.2 V compared to 4.4 V and 4.8 V. As the performance during the thermal test did not show this effect at room temperature, with the higher operating voltages having approximately equal performance to the standard bus voltage of 4.2 V, this degradation must have come from an increased noise output from other components of the bus. Though the degradation is small, with the sensitivity only reduced by about 1 dB between the 4.4 V bus voltage and the 4.2 V bus voltage, the ADS-B receiver operates at very close to its sensitivity limit on orbit. If the performance while on orbit was not satisfactory, it was decided that the bus voltage would be raised
during payload operations to reduce the electromagnetic interference and increase the sensitivity of the receiver. This came with the tradeoff of increased power consumption by the spacecraft, and would have reduced the amount of time that the payload could operate per orbit. Fortunately, the payload performed well once on orbit, so this workaround did not need to be implemented.

It should also be noted that with the transmitter turned on, the ADS-B receiver was severely desensed by around 20 dB. As a majority of the operations were scheduled to take place over the North Atlantic Ocean where the transmitter would be powered off, this would have had only a minimal effect on the primary objective of the payload. It would only become a factor if additional operations were scheduled over North America, where the SFL ground station would make contact with the spacecraft during ground passes and turn the transmitter on. Operations could, however, be conducted over North America relatively easily by reducing the number of ground passes during which the spacecraft was contacted each day.

7.4.2 Thermal Vacuum Testing

During thermal vacuum (TVAC) testing, the spacecraft was tested at the system level in an environment as closely resembling space as is possible. SFL’s thermal vacuum chamber is a large cylindrical metal chamber, with a nitrogen shroud to simulate the coldness of the space environment, and heating lamps to simulate the sun and keep the spacecraft from falling below its survival temperature and being damaged.
At the beginning of the test, the chamber was pumped down to $10^{-5}$ Torr with the spacecraft powered off. Once the chamber pressure reached this level, liquid nitrogen was flowed through a matte-black nitrogen shroud that wraps around the interior wall of the chamber and the heating lamps were powered on. Once the temperature of the satellite stabilized at near room temperature, the TVAC test was begun. The TVAC test followed a similar profile to the one shown in Figure 7.9, but with simulated orbits replacing thermal cycles.

During the TVAC test, only system-level tests were performed on the spacecraft, as opposed to full unit-level functional tests. In general, these tests are as close to unit-level LFFTs as possible, but considerations are made to simplify some tests due to constraints imposed by the spacecraft and the test setup. The high-level test setup for the ADS-B receiver portion of the test was very similar to the EMC test shown in Figure 7.11, only with the thermal vacuum chamber replacing the anechoic chamber. During the thermal vacuum test, a full LFFT was run on the ADS-B receiver with the exception of the sensitivity sweep. It was decided to skip the sensitivity test for two reasons. It would have been difficult to calibrate the input power at the receiver in the same manner as during the EMC test, and the reflections off of the interior walls of the chamber could cause unwanted multipath interference, something that pulse-position modulation is especially prone to. Instead, messages were transmitted in the same manner as the EMC test but at a higher power, in order to verify the functionality of the receiver rather than explicitly measure its sensitivity. This approach is similar to that used for other receivers that have flown on previous spacecraft [18].

![Figure 7.13: The CanX-7 spacecraft in the SFL TVAC chamber](image)

During the course of the test, the functionality of the receiver was again verified at spacecraft temperature limits while in a vacuum. In addition, several orbits of simulated operations were performed. The ADS-B receiver encountered no issues during any of the testing, and all functional tests were passed successfully.

### 7.5 Conclusion

The CanX-7 ADS-B payload hardware had to pass several unit-level and system-level tests before it was accepted for flight. The software functionality and the receiver sensitivity were tested under various operating temperatures and voltages. In addition, the hardware was tested both on its own and while
integrated into the spacecraft in a space-representative environment. Based on its success in passing all of the unit-level and system-level hardware tests, the hardware was accepted for flight on the CanX-7 spacecraft.
Chapter 8

Conclusion

Launched on September 26, 2017, the CanX-7 mission has been a great success. The ADS-B payload collected aircraft data from October 2016 to April 2017, receiving over 4.5 million ADS-B messages in that time. At the conclusion of ADS-B payload operations, the drag sails were successfully deployed to begin deorbiting the spacecraft.

This thesis has presented the work performed to take the CanX-7 ADS-B receiver payload from a prototype to a fully flight-qualified and operational payload. The simulation technique presented was used to determine the requirements for the spacecraft antenna and demonstrate that with the antenna that was designed, the payload would be able to cover a wide swath area, meeting its requirements to track aircraft over the North Atlantic Ocean.

The design and testing of a circularly-polarized patch antenna was also presented. The patch antenna design had to meet the payload’s needs while being constrained by a set of system-level spacecraft requirements. The resulting design was small, only 75 mm x 75 mm, had moderate peak gain, and a wide beamwidth, allowing it to cover large portions of target areas at once. The antenna was circularly polarized without the use of active electronics, a combiner, or multiple feed ports. The antenna was also tested in an anechoic chamber, where its gain pattern and polarization were measured.

The payload’s hardware and software were optimized to increase functionality, robustness, and flexibility. Hardware modifications were made to remove unnecessary components, while software was written to take maximum advantage of pre-existing features, minimizing development time. Underclocking of the payload computer, along with the removal of the unnecessary hardware components, resulted in a payload that consumed less power and generated less heat than the prototype, allowing it to operate for longer periods of time and to collect more data than was originally planned. Despite using less power, the software that was designed introduced a pseudo-bootloader, a read-only storage partition, and a new boot sequence, increasing robustness by introducing a non-modifiable safe state for the payload computer, which would allow for file transfer and new code updates while on orbit. This functionality was tested while on orbit and was successful in improving the time-tagging performance of the application software. Numerous other improvements were made to the software, including the addition of temperature monitoring functionality, an improved telemetry system, and multi-threaded recording of ADS-B data, allowing the payload to respond to and act on commands from the spacecraft at any time.

Lastly, the payload hardware was tested thoroughly to ensure that it could stand up to and function properly in the space environment. Workmanship quality of the payload components was assured through
thermal shock testing. The receiver sensitivity was tested and confirmed to be stable across its maximum operating temperature range and a range of operating voltages. The payload was then integrated into the spacecraft, and tested for electromagnetic compatibility with the rest of the avionics in a flight-representative state. During this test it was confirmed that with the spacecraft avionics in a state similar to that under which payload operations would take place, the receiver was not significantly desensed by the spacecraft avionics. The integrated spacecraft and payload also passed thermal vacuum testing.

With the CanX-7 secondary payload operations concluded, it has been demonstrated that the payload was an unqualified success. The payload was able to track aircraft across the entire planet during its seven month mission. A successful demonstration of a space-based ADS-B receiver is the first step to enabling many types of aircraft tracking missions. With the CanX-7 payload having successfully demonstrated its ability to track aircraft from space, future SFL missions have a solid foundation to build upon.
Bibliography


[38] Nav Canada. AIP Canada (ICAO) Part 2 - En Route, August 2017.


