Final Report for the
StratoPigeon Data Storage and Delivery Capsule Experiment
Flown on the High Altitude Student Platform 2010 Balloon Flight
August 2011

Connie Ciarleglio
David Thoerig
Kristy Weber
Chris Carlsen

Table of Contents

Introduction

The University of Maryland StratoPigeon team is pleased to present the final report for the 2010 UMD payload, a supplemental data storage and delivery system for Antarctic research payloads. The University of Maryland has participated in a total of 3 years of the High Altitude Student Platform (HASP) program, including 1 year of participation by the StratoPigeon team. The StratoPigeon team tested a prototype design during the HASP flight, including software, electronic, mechanical, and ground support subsystems. The StratoPigeon payload was designed and tested in the Aerospace Engineering department at the University of Maryland in College Park, with funding and support from the Maryland Space Grant Consortium and overhead covered by the UMD Space Systems Lab.

Overview

In recent years, Antarctic scientific ballooning flights have become increasingly long as the technology and methodologies used for high altitude ballooning have become more advanced. Ultra Long Duration balloon (ULDB) flights are in excess of 3 months, providing increased environmental access for scientific researchers. The majority of research payloads are composed of heavy, and sometimes delicate, scientific instrumentation, in addition to power and other vehicle hardware. Payloads are almost always above 3,000 lbs. In addition, almost all payloads collect large amounts of scientific data, well into the terabytes range, over the course of their 14-100 day flight.

In order to collect data and verify payload functionality, payloads normally use the TDRSS satellite link at a maximum transfer rate of 100 kbps. While the link is good for slower data, the majority of the scientific data is stored on board. Even with a direct line of sight radio link, of approximately 333 kbps, data transfer would occur comparatively slowly. Due to their size and weight, the recovery of research payloads is not always an easy task. Many plane trips and experienced personnel are usually required for payload retrieval. Sometimes, due to weather and landing location, the payload must be retrieved days or weeks later or, rarely, not at all.

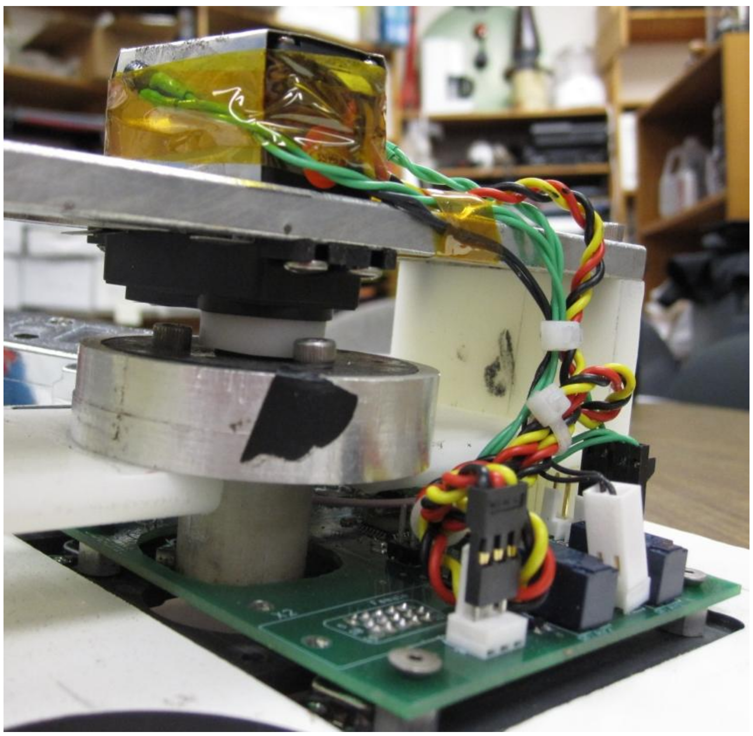
The StratoPigeon capsule is designed as a complementary system to the TDRSS and on board data storage system currently in place for Antarctic research payloads. The capsule has a capacity for up to several terabytes of data, can land in an easily accessible location, and is lightweight and small for single person retrieval. On ULDB flights, several capsules can be launched with the main payload and released near base and data retrieved easily during flight. It can also be used as a backup storage system and helps to minimize risk associated with main payload retrieval. During the HASP 2010 flight, a prototype capsule was flown and retrieved to test flight subsystems and ground operations.

Flight Systems

Mechanical Systems

Final Design

The mechanical system for the capsule consists mainly of the detachment mechanism for the payload from the main gondola. The system is a simple servo-driven mechanical decoupling device. The servo is mounted to a reinforced aluminum bracket on top of an interface plate permanently attached to the main gondola. The detachment mechanism is a single bolt action lock, consisting of a delrin horn with a locking pin attached to the servo, and a steel sheath attached to the payload. The locking channel cut into the sheath has a 90 degree turn cut into it, requiring no active power to maintain a positive lock. To detach from the interface plate, the servo simply rotates 90 degrees and a small compression spring, mounted inside the sheath, pushes the payload away, ensuring detachment from the electrical interface connectors.



The parachute, made from a high visibility orange nylon material, is attached to all four corners of the payload and deploys passively upon detachment. The body of the parachute is folded into a clear acrylic tube which is mounted on top of the interface plate, to allow visual confirmation of deployment by cameras onboard the gondola.

The structure of the payload consisted of a modified commercial container for PC104 hardware. The container was 7” tall and 1/8” thick aluminum. While the container adds significant weight to the payload, it helps to ensure that the electronics remain intact on payload landing. The electronics were mounted on a central stack attached to the top of the payload. PC104 mounting specifications were used for ease of integration. The battery was mounted with an aluminum bracket to the bottom plate of the payload to ensure that the center of gravity of the payload remained low inside the capsule structure.

Prior to the HASP flight, a series of drop tests were performed to verify the functionality of the mechanical systems and to ensure a completely vertical drop trajectory for the payload. The mechanical system performed successfully in all of these tests. During the HASP flight, the functionality of the mechanical system was observed through live video feed from CosmoCam. The servo system performed as expected during flight with no appreciable deviation in trajectory from previous trials.

Due to protrusion of the antennas from the bottom of the payload, a crush pad system was implemented to prevent antenna damage on landing. The chosen material was a simple cardboard honeycomb structure which covered the entire length of both rubber duck antennas. The structure added minimal mass and worked well for antenna protection. On examination of the payload at the landing site, the crush pad had a single minor dent in one corner but was otherwise intact.

Mechanical Design Issues

During initial testing it was found that the servo was vulnerable to damage during landing, pushing the servo horn into the body and shattering the internal gears. The servo horn mount was reinforced to prevent this from reoccurring.

Additionally, an issue was found with the mounting procedure for the payload. The spring internal to the release mechanism and electrical interface connectors requires a fair amount of force to overcome, requiring three people to correctly load the payload on to the interface plate. Alternate methods of attachment are now being discussed to make attachment a simple, one person job.

Electrical Systems

Final Design

The electronics subsystem is composed of a central PC104 stack, 2 Atmega based custom printed circuit boards (PCB's), and a programmable GPS and radio communications system. The PC104 controls data transfers and has attached solid state drives (SSD's) for data storage. The first Atmega board is used for payload detachment and connects to the servo in the mechanical system. The second Atmega PCB located internal to the capsule is used for power distribution and regulation from the main HASP 30V bus and the internal 12.8 V battery pack.

The detachment system electronics are based around an Arduino microcontroller. HASP serial data is distributed to both the servo microcontroller and power microcontroller through the use of RS232 multi-drop chips. The servo electronics control both actuation of the servo mechanism and the heating mechanism for the servo. Serial, Ethernet, and I/O lines are passed from the servo PCB to the power PCB internal to the capsule via 2 high density DB15 connectors. The board is mounted directly to the payload plate with the servo pin mechanism mounted internal to the PCB layout. The servo electronic board layout is shown in the appendices.

The power distribution electronics for internal capsule components are also based around an Arduino microcontroller PCB. The PCB uses a series of relays to control power to all components internal to the payload. The board also utilizes an RS232 multi-drop chip to share a single RS232 line with the servo electronics. Diode protection is used to ensure that HASP and internal battery power do not interfere with one another. The power distribution board layout is included in the appendices.

The final communications and tracking electronics consisted of a 2 radio and GPS configuration. The first tracker was a reverse-engineered TinyTrak by the company Byonics. The transmitter was connected to a Garmin GPS15x with an active antenna mounted to the top of the payload. The second transmitter consisted of the Xtend radio by Digi which has been used and tested by UMD teams in past balloon flights. This radio was connected to a pre-programmed Lasen IQ GPS, also with an active antenna mounted to the top of the capsule. The Lasen IQ was programmed to transmit at a matching interface baud rate to the radio and at a rate of once per 15 seconds.

The main computer used for data storage and transfer was a Vortex86DX based PC104 processor with an attached SATA expander board used to connect the 3 internal SSD's. The PC104 board runs at 800Mhz and is rated for industrial temperatures. The low cost of this CPU makes it ideal for use in the prototype capsule. The SATA expansion board allows up to 4 SATA devices to be connected to the CPU. 2 industrial rated SSD's and 1 Intel commercially rated SSD were attached to the main for the HASP flight. Ethernet to the main CPU is transferred in through the servo electronics and into the payload.

Flight Anomalies

There was one anticipated anomaly with the Garmin GPS15x during the portions of flight where the tracking system was active. As demonstrated in previous HASP flights, firmware for the Garmin 18x, the same firmware as the 15x, cannot acquire a lock at altitude above or close to 120,000 ft, despite acquisition of sufficient satellites for a solid lock. The GPS will reacquire a lock at a lower altitude. This was anticipated as a problem when the communications was designed, especially for the immediate time period before detachment of the payload from the gondola and during descent. The Lasen IQ GPS was chosen as the companion GPS to the Garmin for this reason, as it is flight proven at 120,000 ft and above. While only a single GPS had a good lock before descent, with the knowledge that a lock on the second GPS would be acquired at lower altitude, the risk was deemed acceptable for the payload. Actual flight conditions matched those described above and the payload was released with a single GPS lock. The Garmin reacquired a lock at approximately 60,000 ft while the Lasen IQ held a lock throughout the descent portion of flight.

Electronics Design Issues

During the course of electronics development, there were 2 major issues, resolved prior to the HASP flight that posed a problem for the flight hardware. Power distribution, particularly in a system with 2 voltage buses and power sources was a challenging problem. The diode configuration developed let HASP voltage pass so long as the HASP bus was above the battery bus voltage. When HASP voltage was turned off, as prior to detachment, the battery voltage was allowed to pass. In this configuration, the battery is only capable of powering the internal payload, and not the servo electronics.

In the few weeks before the flight, an anomaly was discovered such that the battery voltage was passing up into the servo board and on to the HASP voltage lines. As this indicates a short or leak somewhere in the electronics and could be potentially damaging to HASP and when the HASP power is inactive, it became a flight critical issue. It was determined through testing that the problem was not a direct short, but a leak somewhere in the board. Even with a minimally populated (2 or 3 necessary components) power distribution board, the problem still occurred. While the problem was never conclusively solved, adding a diode into the power line of the servo electronics in the opposite direction than was anticipated seemed to mitigate the issue. Further investigation is required to fully flush out the problem.

A flight critical issue was also discovered with the communications PCB in the weeks prior to flight. The original communications design used an Arduino based microcontroller mounted to a PCB with an Xtend, TinyTrak, and 2 Inventek GPSs. During integration, both transmitters were successfully tested, but the GPS units could not acquire a lock. The problem was later discovered with an oscilloscope attached to the serial lines of the GPS. When either radio made a transmission, there was a large amount of noise on the GPS serial lines. The noise included voltages above the 3.3V rating of the GPS and negative voltages, rendering the GPS unusable. It was concluded that the design could not be remedied in time, so a new final design was developed, as described in the final flight electronics section above. The new design is actually much simpler and easier to use than the original.

Software System

Flight software for the 2010 payload consisted of 2 microcontrollers programmed with Arduino and a main computer stack with a custom Debian OS and bash shell script for testing of data transfers. The Arduino programs are a simplified version of C++ that use object oriented programs to describe relays and attached devices. Flight software on the servo PCB creates 2 data strings containing the state of the servo relays and the both servo and ambient temperatures. Flight software on the power distribution PCB also creates 2 serial strings containing power distribution relay states and internal payload temperatures. I/O lines between the 2 microcontrollers are used for transmission timing, as both microcontrollers cannot transmit to the single RS232 line at the same time. Each microcontroller also toggles an I/O pin to activate or deactivate the transmit capability on the RS232 multi-drop chip. Each microcontroller runs at a clock rate of 8 Mhz. Serial transmissions for both microcontrollers were composed of the following:

String	Length	String Elements
Servo Relays	30	Start Identifier, element identifier, relay state, endline
Servo Temperatures	19	Start identifier, element identifier, temperatures in Celcius, endline
Power Relays	71	Start identifier, element identifier, relay states, endline
Internal Temperatures	21	Start identifier, element identifier, temperatures in Celcius, endline
Debug	62	Debug characters

A custom Debian board package was used for the main PC104 stack. In addition, the boot files were modified to recognize the attached solid state drives on boot up for the purpose of identifying areas for file transfers. Thus, each SSD would show up as the same numerical name for a communicating remote device. A short bash shell script was used to test the transfer of several 1.2 Gb image files on system boot up.

<u>Flight Anomalies</u>

In payload testing, the phenomenon occurred sometimes that when a relay command was sent, the command would not be executed. The issue was tested and errors in the commanding association with relay numbers was determined to be the issue. (I.e. There was a mix up in the relay commands for which relay was which). However, it was also noticed that sometimes when the payload was restarted several times, the issue seemed to resolve itself. However, with the command confusion, the problem was determined to be fixed enough for flight. During flight, the internal heater command was sent at least once from the base station. However, recorded temperatures and relay states indicate that the command was not sent, in much the same situation observed during testing. The command list was checked prior to flight for verification assurance. The issue was never fully resolved but did not appear to affect payload operations during flight.

<u>Software Design Issues</u>

One of the positive aspects of using SATA drives for data storage is that each drive can be addressed individually by its UUID and set to a specific mount name. This makes sure that each drive is mounted the same after each power cycle of the computer. One problem that was discovered was that while this method is beneficial, it also causes problems in the boot cycle if the drives are unpowered when the computer starts up. If the computer attempts to boot with UUIDs configured in the boot scripts and the drives are not attached or unpowered, it requires keystroke input to bypass the error thrown when no drives are recognized. As the system is remote, this would not be possible on a high altitude flight. If this error occurred during flight, the data transfer portion of flight would be considered a failure. This issue was not a problem during the 2010 flight but will be improved for future revisions.

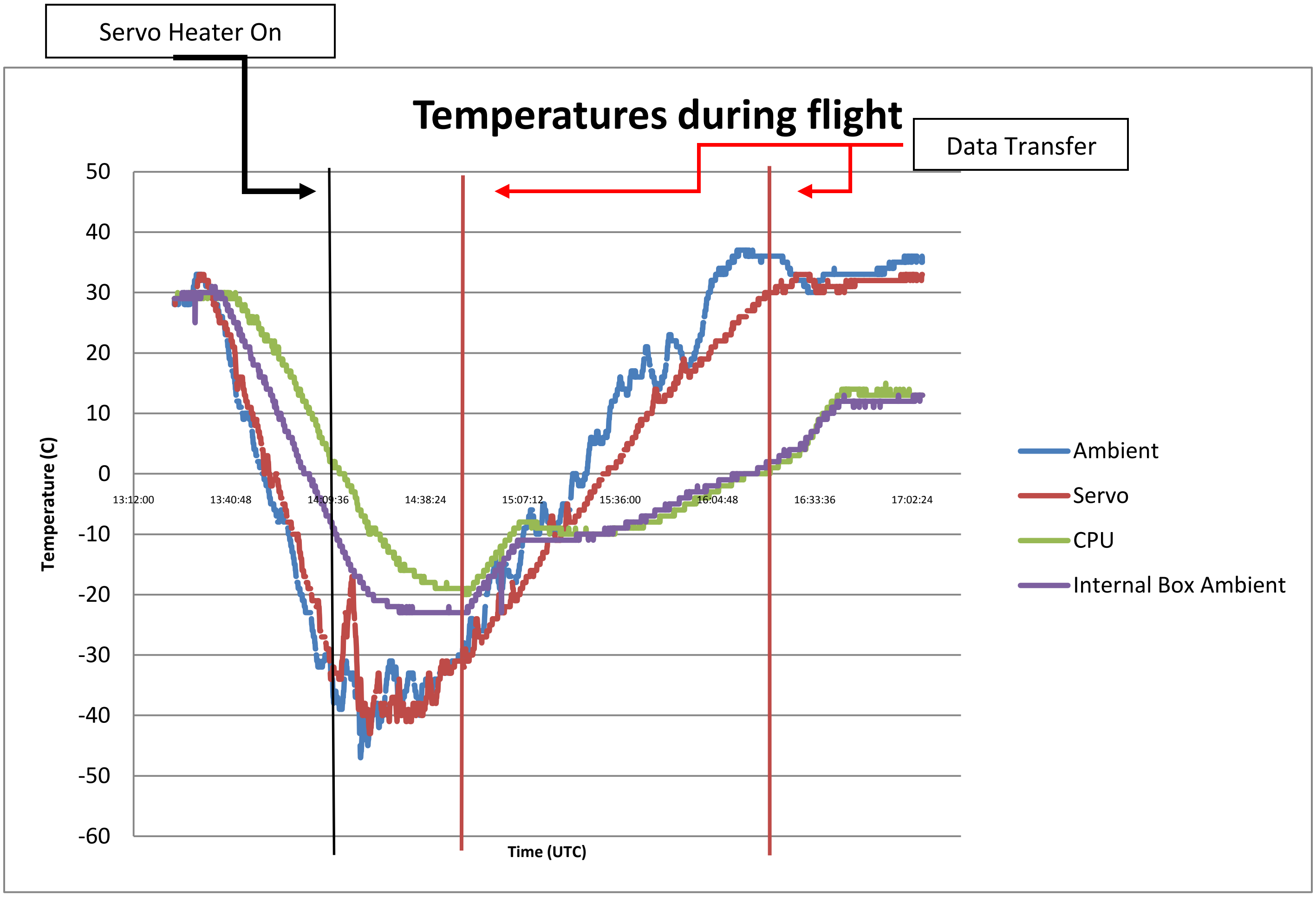
Thermal Management

The thermal management system for StratoPigeon is a combination of passive and active components. Mitigation techniques include the use of heat sinks, straps, and thermal control coating on hot components and heaters and housings on cold components.

Because the system can draw close to 15 W during high use operations internal to the capsule, heat sinks were required on all hot components. The outside of the capsule was painted with high emissivity white paint and many hot components were sinked to the aluminum enclosure. The CPU had the highest power emmitance and a custom copper sink was built to attach the CPU to the side of the enclosure. Each SSD also had an aluminum heat sink attached to the front, as well as being attached to the enclosure on the side.

To endure the cold portions of flight, heaters were included internal to the capsule and attached to the servo. Each heater was made of Kapton and covered a 1" square area while emitting 5 W of power. In addition to these techniques, the majority of the components in the payload were industrial rated at -40 – 80 °C. The commercially rated components included the commercial SSD for temperature testing purposes, the SATA expansion board on the CPU, and the servo.

Thermal data was collected during the flight and is shown in the following graph:



Sensor	Minimum (°C)	Maximum (°C)
Ambient	-47	37
Servo	-43	33
CPU	-20	30
Internal Capsule Ambient	-23	31

Hot case thermal control worked very well. The maximum temperature encountered was well within commercial component temperature ratings. Cold case thermal control was only partially successful. As mentioned previously, the internal capsule heater actuation method failed and no additional power was used to keep the internal payload warm. However, when the CPU and SSDs are active, almost 15 W is generated, which is enough to keep the internal payload minimally warm. However, as seen in thermal chamber testing, CPU heat is not sufficient if the payload is held constantly at low temperatures. For commercial components inside the capsule, additional heating will be required, especially considering that the internal capsule spent the majority of the time below 0 °C.

The servo heater was very minimally successful. While turning on the heater did increase the servo temperature significantly, when the heater was shut off, the temperature dropped rapidly. A less powerful heater that can be turned on for long periods of time, without risk of burning the servo, would be beneficial.

Ground Systems

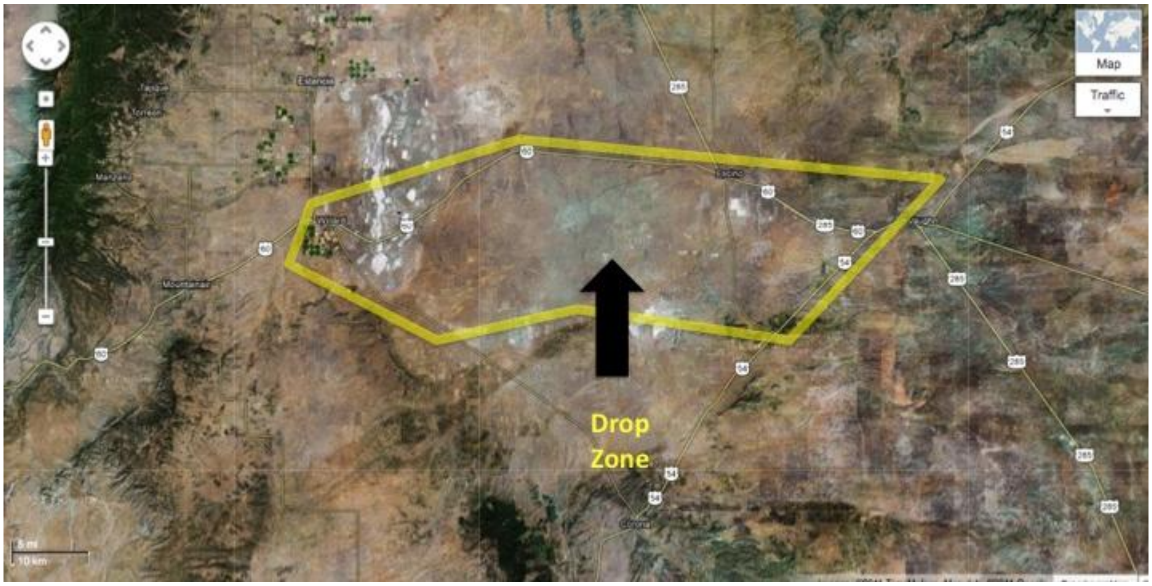
Flight Path Prediction

Balloon Track software used data from NOAA daily radio soundings of the upper level winds to predict the heading and location of the payload ground track. The software utilized characteristics such as drag, ascent rate, and descent rate in order to approximate the payload's location while passing through the different layers of the atmosphere. Balloon Track also had the advantage of outputting the flight path onto a map using the program Microsoft MapPoint. This allowed for the tracking team to find a proper location in HASP's flight trajectory that would allow the payload to land in an easily accessible location, also while avoiding potential flight hazards such as airports, heavily populated areas, and bodies of water.

Days before the flight, soundings taken at the NASA flight facility in Fort Sumner, New Mexico, were used to get a general idea of the wind patterns typical for this portion of the southwestern United States during late August. This data was also used to familiarize us with the terrain that our flight would potentially be covering, and to plan potential routes for flight recovery.

Using data from a Pi Ball released the morning the flight at the NASA flight facility in Fort Sumner, we converted all the necessary sounding data into a format suitable for Balloon Track to work with. A number of predictions were made in order to find a number of suitable points along the predicted float path of HASP 2010 from which the StratoPigeon release command could be made.

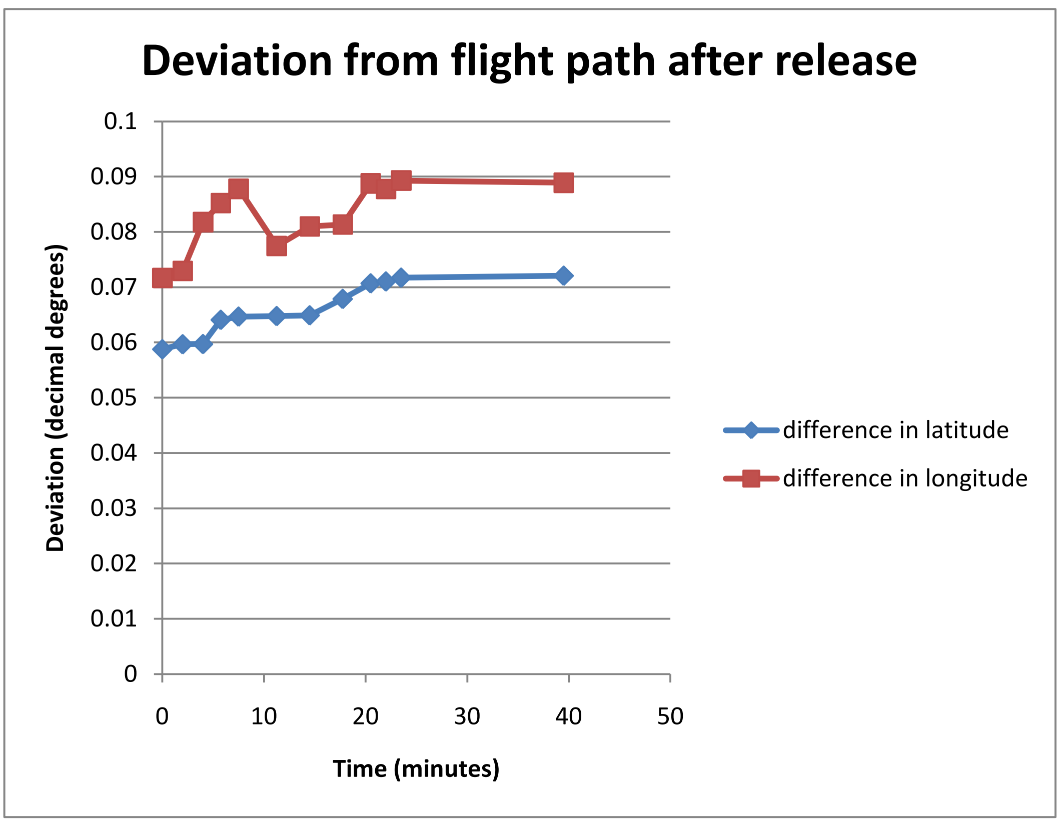
Before HASP was launched, the team conferred and agreed on a "safe" region in which the drop sequence command would be given, which is the region inside the yellow shape outline in figure 1. This is a region north of Duran, North and east of Progresso and bordered by highway 42, and west of I-54. Because it was likely the actual flight path would deviate somewhat from the predicted path, the payload was planned to be released as soon as HASP got close to the 105.5th parallel.



Predicted Drop Zone

Upon release, StratoPigeon fell along the predicted semicircular path, but with small deviations, as can be seen below. The predicted descent was traced in yellow, while the actual descent is in green. These deviations were by about 0.059° to the north, and 0.071° to the west. These deviations from the predicted path grew to about 0.072° to the north and 0.089° to the west by the time the payload landed, giving the payload an average deviation from the predicted flight path of 0.066° to the north, and 0.083° to the west. The growth of these deviations during descent can be seen in the flight path deviation graph below. This stronger deviation to the west can probably be attributed to the same strong easterly wind which caused HASP to move west much more quickly than anticipated, giving StratoPigeon the shorter than expected flight time.

Predicted and Actual Descent Trajectory



Flight Path Deviation

Base Station Operations

As soon as HASP 2010 was launched, we began to receive packets that monitored temperatures of a number of important locations within StratoPigeon. This data was used to make sure that different components would not go beneath their functional temperature, and would remain operational throughout the entire flight. This was especially important for the servomotor and CPU, as the heaters needed to be activated if the temperature got close to -20°F, so that the servomotor would not be frozen during the drop procedure, and data transfers would still be able to be made. Monitoring temperature was also vitally important during ascent, as the payload rose through the tropopause, which was the coldest portion of the flight, and had the potential to damage components even though the payload was only in this region for a short period of time.

Communications were kept regularly with the StratoPigeon tracking team using HAM radios on a 2–meter band with the Mega-Link repeater network in New Mexico. This way, the base team and recovery team would be regularly updated with the location of the payload, so long as at least one of the teams was within range of the payload’s signal and network. Before the drop sequence started, alert was given to proper officials at the NASA flight facility, in order to follow with safe FAA procedures. Communications via HAM radio were also vitally important once StratoPigeon was released from HASP, because this was the only way the payload could be tracked by the recovery team. As the payload descended through different layers of the atmosphere, its latitude, longitude, and altitude were relayed back to the proper officials so that the necessary airspace could be safely cleared for the payload, until it was confirmed that the payload had landed on the ground, or passed through a safe altitude.

One of the difficulties of tracking StratoPigeon was that it was very difficult to determine the actual ground speed of the payload. The prediction software we used only gave latitudes and longitudes, which could, after some calculations, be converted to the track covered by the payload. However, there were multiple parts of the flight being monitored all at the same time, and this would have been too time consuming. The software, however, did list elapsed time along with the prediction of the position of the payload at this time. This way, we were able to monitor how closely the payload stayed on its flight path throughout the entire flight. This flight allowed us to realize the ever-changing conditions of the atmosphere, as we unexpectedly hit a streak of air which caused the payload to cruise at a much higher than expected velocity at float, causing the time at float to be much less than predicted, and a more rushed drop sequence to be undertaken.

FAA Integration

Communication with the FAA was performed through Bill Stepp. Actions and status of the StratoPigeon team and payload was relayed through Bill Stepp to the regional FAA controller. Bill Stepp, before the launch, set out procedures in which the prime recovery zone and an estimated detachment time for the payload were pre-determined and sent to the FAA the morning of the HASP flight. After discussing feasible and allowable recovery zones with Bill Stepp, a good predicted path for HASP was required before proceeding to ensure that the payload could be recovered in a populated zone where the CSBF team was familiar with the area. We also received an airspace sectional map of the area to assist us in determining a “good” cut down location based on the current flight trajectory. Along with lat/long coordinates, the sectional map had airport zones and "no fly zones" indicated. This map was used to determine a proper location both in terms of recovery and flight path. Because of mountains, national forests, and reserve property we choose a closer than originally desired recovery location to insure the highest probability for a timely recovery in a well known area.

The planned flight procedure was to provide a half hour warning before payload release to allow ample time for FAA notification through Bill Stepp. The course of the actual flight did not follow the predicted path completely so both corrections in landing zone and recovery team location delayed the warning of payload release to only a few minutes. After release, the tracking team relayed the location and altitude back to mission control which in turn gave the information to Bill Stepp who passed it on to the FAA. Because of these frequent updates, the lack of pre-release notification time was not a problem. During the descent of StratoPigeon there was little difficulty relaying the information to mission control through cell phone. While the 2m repeater network was used to relay information the majority of the time before payload release, cell phone coverage worked well for the descent portion of the StratoPigeon flight.

Tracking Team Operations

The tracking team consisted of a primary tracker and a driver, also serving as a backup navigator. The plan for recovery was to send the recovery team out ahead of the balloon and modify the course based on navigational information relayed from the base station. This was to be done through ham radio with cell phones being the back up. The tracking vehicle was outfitted with two ham radios, one for receiving APRS packets from StratoPigeon, once its GPS was turned on, and the other for voice communication with mission control.

After launch, one of the immediate problems involved communication over the 2m repeater network with the base station. For increased range of communication, the 2m tracking radio was tuned into a repeater network which allowed the recovery team to reach mission control from otherwise out of range locations. What was unforeseen was the sudden use of this repeater by the local population. This delayed communication on several occasions but was not a major issue. At first the conversation ranged from old stories to tomato gardens but once the local ham operators realized that tracking operations were in progress, the network was largely cleared. It is not anticipated that this will be an issue for Antarctic operations.

The tracking team was unhindered by traffic during the ascent, float, and descent portions of flight and was able to navigate roads quickly. An issue occurred partway through tracking with the 900 MHz Omni antenna. As the antenna was removed from the radio module in or to insert the yagi antenna, the tubing in the female RP-SMA came out of the connector. It was thought, at the time, that the pin had come out of the male side of the connector, located in the main radio module. If this had been the case, the 900 MHz signal would not have been receivable as no backup module was packed for the tracking segment. As it was, the 900 MHz Omni was unusable but the directional antenna was more than adequate for signal reception for the remainder of the tracking period, if requiring a significant amount of hand-pointing.

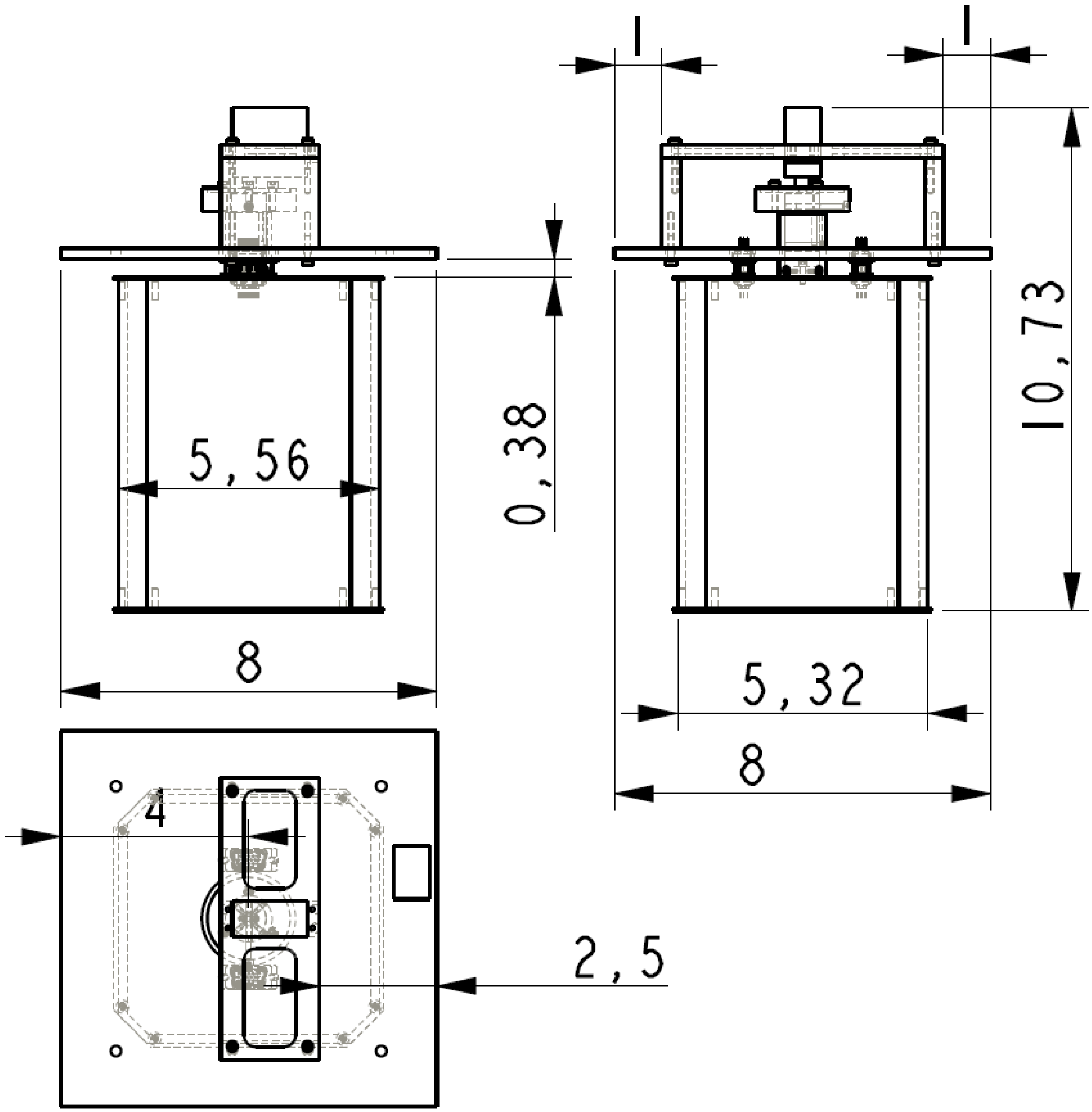
Also, an unpredicted air current which pushed the balloon off the predicted path led the recovery team to rapidly alter their position to be in range of the signal upon release. Once the command was given and StratoPigeon was released the recovery vehicle was parked and the secondary tracker relayed information back to base station and watched the 144.39 signal while the primary tracker made certain the Yagi antenna was pointed correctly and recorded data from the 900 MHz beacon. Once StratoPigeon dropped into a low enough altitude that a rise obstructed the signals from reaching the recovery team, the team set out to move closer to the last known coordinates and reacquire the signal. After navigating to a closer road and traveling it for several miles, the ground signal was obtained and the landing location of StratoPigeon was determined. The tracking team then received a phone number for the land owners and after several phone calls, the proper land owners were reached and permission was given to enter the property and retrieve StratoPigeon from its landing location. Contacting the land owners was somewhat lucky but permission to enter the land was given very readily.

Once on the property the recovery team spotted StratoPigeon's parachute blowing in the wind and recovery was completed. The team was not able to shut off StratoPigeon’s beacon at the time of recovery due to lack of correct tools and space.

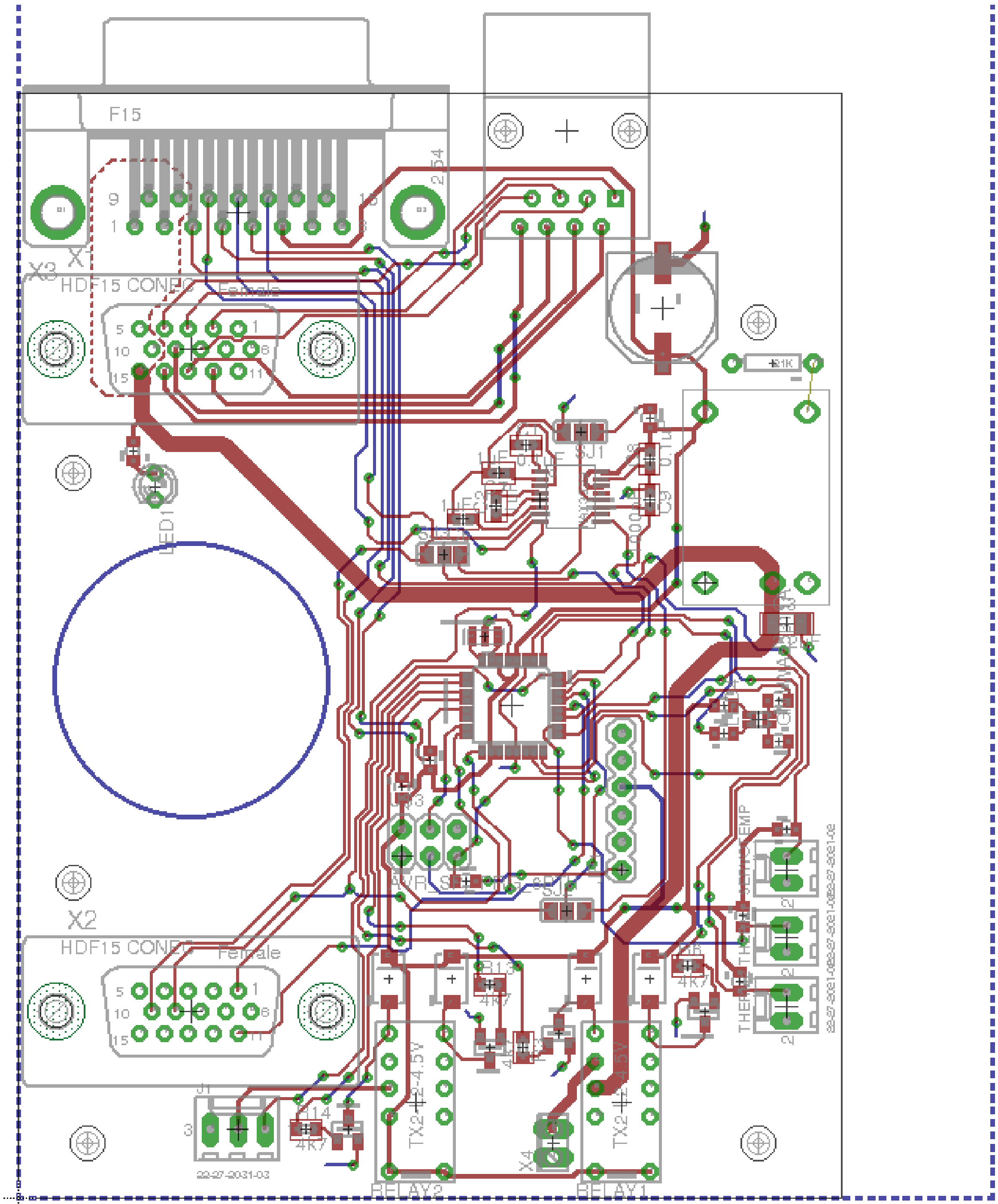
Conclusion

The 2010 StratoPigeon payload provided an effective prototype payload to determine feasibility, methodology, and future usefulness for this type of data storage module. The team feels that we have had sufficient success with internal payload systems and tracking and ground operations to call the prototype a success. The hardware, software, and tracking methods used were a test for a prototype the team hopes to further develop and utilize to aid Antarctic research teams on ULDB flights. The use of this payload opens a great number of possibilities for long duration research missions that require the data storage capacity and ease of recovery of this payload concept. In addition, it adds a low cost backup system to current systems and could be potentially used on any mission with cost, power, and space constraints.

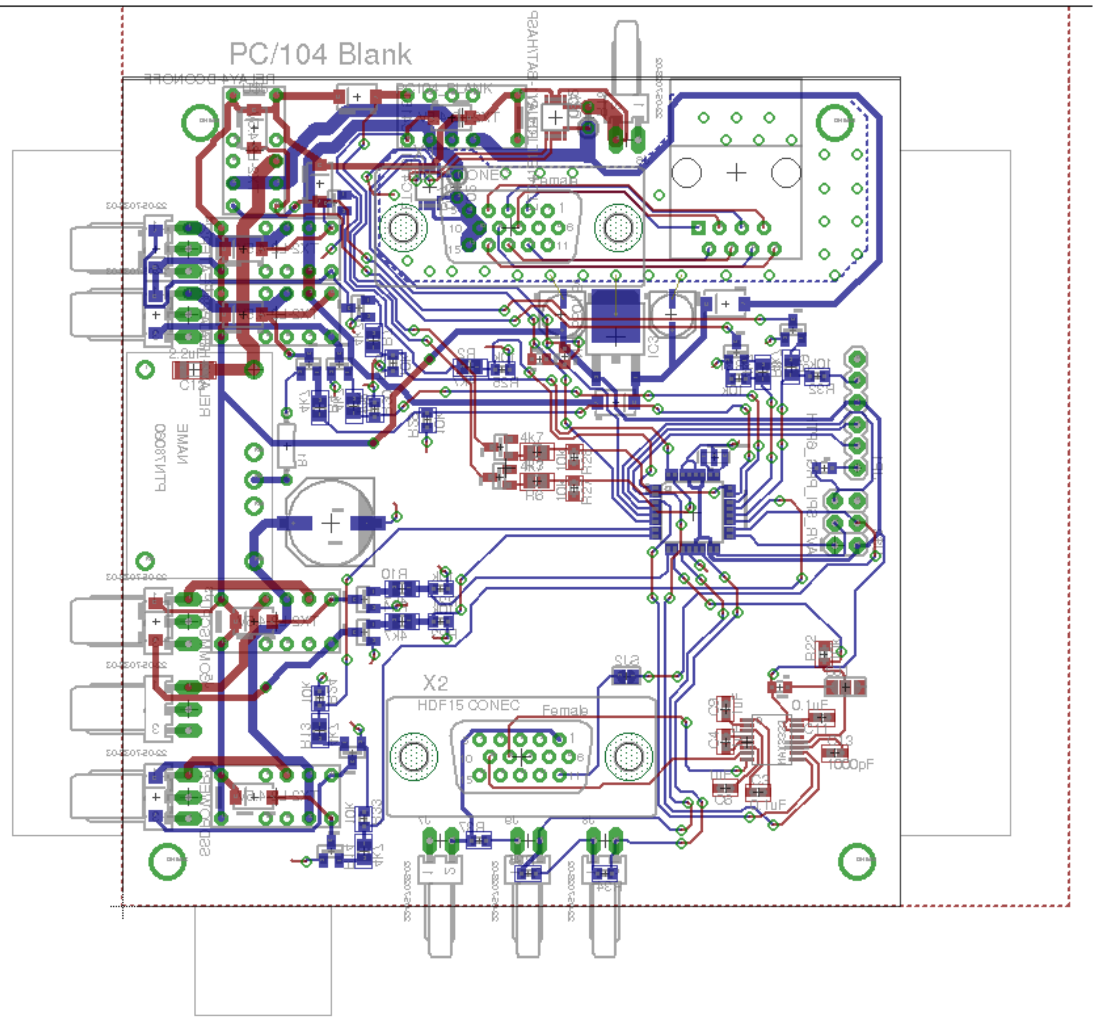
Appendices



Mechanical drawings for the StratoPigeon Payload



Servo Board layout



Power Board Schematic