

PTERODACTYL: A Versatile Flight Computer for Stratospheric Ballooning

Seyon Wallo ^a, Ashton Posey ^a, Paul Wehling ^b, Andrew Van Gerpen ^b and James Flaten ^c

Abstract

In the summer of 2020, the stratospheric ballooning team at the University of Minnesota - Twin Cities developed a versatile flight computer named PTERODACTYL (Payload To Enable Recording Of Data And Communication Telemetry Y (While) Lofted). PTERODACTYL is built on a custom, student-designed printed circuit board (pcb) and is based around a Teensy 3.5 microcontroller. It measures just under 10 cm by 10 cm, allowing it to fit inside a (mock) CubeSat. For a 2 to 3 hour stratospheric balloon flight, a PTERODACTYL can be powered by a single 9 volt lithium battery. It has on-board SD card logging, as well as a short-range XBee3 radio that allows it to communicate with nearby XBee3-enabled payloads on the same stack, such as other PTERODACTYLs units and/or an XBee3-commandable vent. The PTERODACTYL flight computer system has the following built-in sensors: gps (u-blox NEO-M8N), pressure, 9-degree-of-freedom IMU, and two thermistors to measure on-board and external temperature. It also has headers to control two servos, as well as slide switches, momentary switches, and shorting plugs (for user input) plus 4 colored LEDs and a miniature OLED screen (to output information to the user). It is constructed using only through-hole soldering, for ease of assembly. The basic flight code, written to log readings from the basic sensor set, has been used by (and tweaked by) high school teachers, college/university students in freshman seminars and other students new to ballooning, as well as experienced ballooning students. The PTERODACTYL pcb breaks out some of the non-used Teensy pins, allowing easy access to UART, SPI, and I2C buses for more-advanced users. Our team has used the PTERODACTYL flight computer to run RM-60 Geiger counters, other gps modules (most notably the u-blox NEO-M9N), QWIIC sensors (Sparkfun sells a variety of QWIIC sensors, but not all work in stratospheric conditions), real-time clock modules, other radio modules (a second XBee3, an RFD900, an RFM69), relays (to control sirens and heaters), and more. Due to the current difficulty in sourcing Teensy 3.5 microcontrollers, we recently designed a Teensy 4.1 version of the PTERODACTYL pcb and took the opportunity to rearrange the components to make the pcb more user-friendly and to add more functionality. In addition to the new microcontroller, the new PTERODACTYL board can be powered by a 3.7 volt battery or by 5 volt USB input, not just by 9 volt batteries. It is also directly compatible with both u-blox NEO-M8N and NEO-M9N gps modules, which can have different pinouts (depending on which company they are purchased from). The PTERODACTYL has been selected as a flight computer for the "engineering" side (AKA the "video streaming" side) of the National Eclipse Ballooning Project (NEBP). It will allow NEBP engineering teams easy access to "basic" sensor data and also provide a platform to which teams may add custom sensors and/or actuators. Once integrated with the NEBP Iridium modem system, the XBee3 will allow a PTERODACTYL to relay commands from the ground to other payloads on the stack (such as a vent) in near-real-time.

stratospheric ballooning | flight computer

^a Undergraduate student, University of Minnesota – Twin Cities

^b Alum, University of Minnesota – Twin Cities

^c Associate Director of NASA's Minnesota Space Grant Consortium and Contract Associate Professor of Aerospace Engineering and Mechanics, University of Minnesota – Twin Cities

Author to whom correspondence should be addressed: James Flaten, flate001@umn.edu

1. Introduction

Academic stratospheric balloon flights often carry cameras and microcontroller-logged sensor suites to document the view and the environmental conditions through which the payload(s) pass. Data is usually logged to SD cards, or other memory devices, then read back after the payloads are recovered - though some programs send experiment data to the ground by radio telemetry, making recovery of physical hardware and onboard data logs, if any, less critical. Tracking data is nearly always sent to the ground by telemetry, to help find the payloads after they land. Weight, battery life, temperature, and memory are all considerations in payload design, with the goal to be able to collect data from before launch until after landing - often 2 to 3 hours on a typical weather balloon flight that ascends at about 1 m/sec (1000 ft/min), bursts in the lower stratosphere, and returns payloads to the ground by parachute.

The PTERODACTYL (Payload To Enable Recording Of Data And Communication Telemetry Y (While) Lofted - a cute acronym!) flight computer was developed by the MN Space Grant ballooning team at the U of MN - Twin Cities [1] and has been adopted by the Nationwide Eclipse Ballooning Project (NEBP) [<https://eclipse.montana.edu/>] for use by “engineering side” teams around the country during eclipse ballooning campaigns planned for 2023 and 2024.

The PTERODACTYL flight computer is built on a custom printed circuit board (pcb) around a Teensy microcontroller with a built-in micro-SD card slot. It has “basic” sensors (pressure, temperature, and a 9 degree-of-freedom inertial measurement unit (AKA 9DOF IMU)), plus headers to attach additional sensors. It is robust, compact (measuring just under 10 cm by 10 cm, so it could potentially fit into a (mock-)CubeSat). The PTERODACTYL features a gps module which can be set into airborne mode - critical for stratospheric use - so that data from other sensors can be location-tagged. It has switches and a tiny OLED screen for user input/output, as well as a short-range radio module allowing it to communicate with other payloads flying on the same stack. Assembly requires only through-hole soldering. The Teensy microcontroller can be programmed using the Arduino Integrated Development Environment (IDE), so the code is essentially written in C/C++.

2. PTERODACTYL “v2” with Teensy 3.5 (White PCB)

Photographs of the PTERODACTYL with the most flight heritage, known as “v2” or “white” (since it is built on a white pcb), appear in Fig. 1, with the pcb layout, drawn using Eagle software, in Fig. 2. In this section “PTERODACTYL” always refers to the “v2” (Teensy 3.5) variant. Figure 3 shows the parts needed to assemble a PTERODACTYL, which only requires through-hole soldering (an intentional design choice). Figure 4 shows a PTERODACTYL, along with some other sensors and a space ready for a video camera, mounted inside a student-built payload box.

In Fig. 1 the Teensy 3.5 microcontroller, with SD card slot at the bottom, is mounted on plug-in headers vertically, near the center of the pcb. There are four uniquely-colored indicator LEDs (with built-in current-limiting resistors) along left edge, toward the bottom, plus two momentary switches and three slide switches for user input along the bottom edge, toward the left. On the top edge, toward the left, are two jacks for 9-volt batteries wired in parallel (no batteries are plugged into the jacks in the photo). To the right of the battery jacks is an L7805CV 5 volt linear voltage regulator, to step 9 volts from the batteries down to 5 volts for use by other components. (Aside: Using a linear voltage regulator and a single Energizer “Ultimate Lithium” battery has been adequate for “standard operations” with the PTERODACTYL, even in the stratosphere where heat dissipation can be an issue.) The green terminal block on the top edge has a spot for a 9 volt battery direct wiring without a plug, if space is at a premium (as would be the case if trying to abide by a

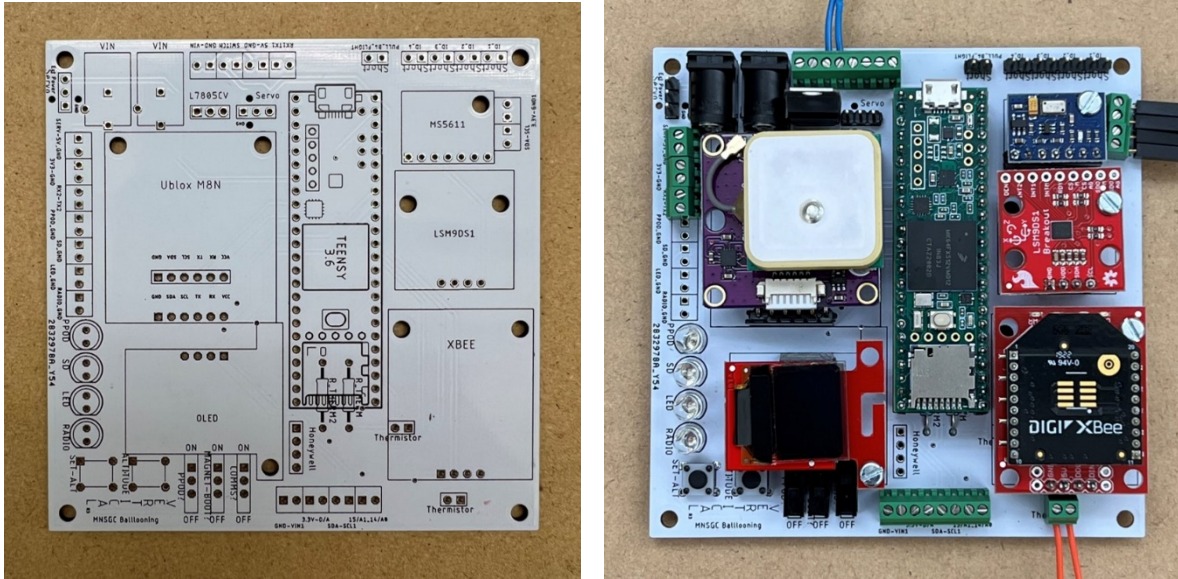


Fig. 1: Bare and populated pcb for the Teensy 3.5 version of the PTERODACTYL flight computer. This pcb is for a Teensy 3.5, despite the board lettering that says Teensy 3.6.

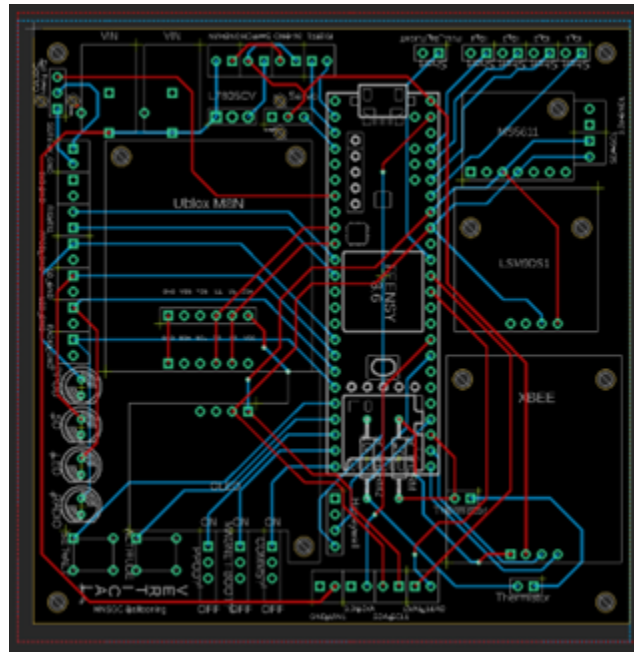


Fig. 2: Wiring layout, drawn with Eagle software, for the custom PTERODACTYL pcb (Teensy 3.5 version).

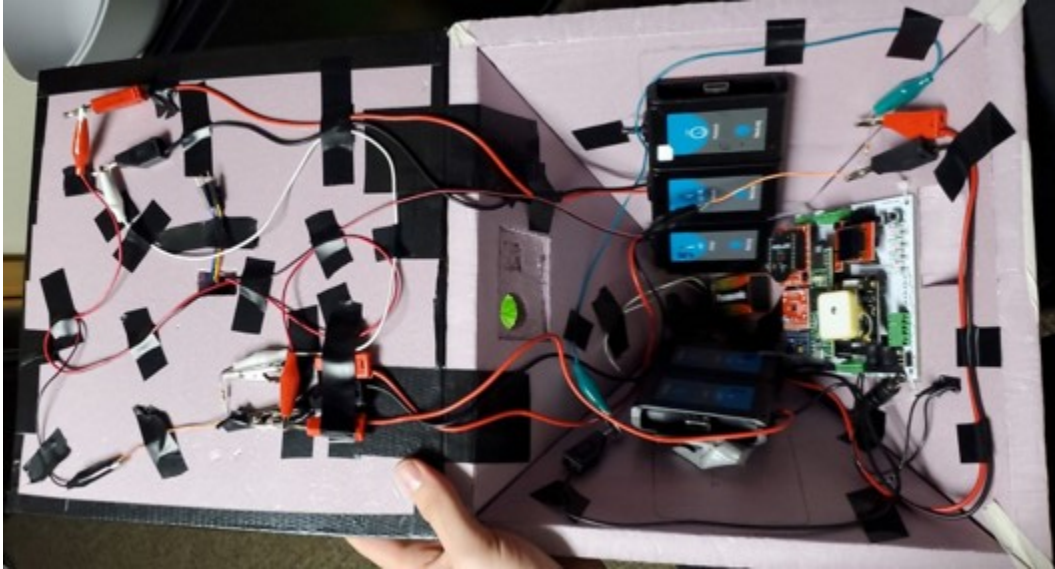


Fig. 4: A PTERODACTYL unit, along with several other sensors, mounted in a pyramid-shaped, student-built payload (right side) with lid open (left side), ready for a stratospheric balloon flight.

CubeSat form factor). The board can also be powered directly by 5 volts through the Teensy power/programming microUSB plug, or by a 3.7 volt battery wired directly to the Vin pins on the Teensy. The voltage regulator on the Teensy is rated for 1 A and the PTERODACTYL, running only its native sensor set at 1 Hz, only draws about 0.33 A.

The blue wires exiting the board from the top going upward in the photo lead to an external switch, which can be played on the outside of the payload box. Four pairs of shorting plugs on the top edge toward the right can be used to indicate unique XBee radio ID codes. There is an on-board PS103J2 10 kOhm thermistor to monitor pcb temperature (under the XBee breakout board, so not visible in the populated board photo), and the red wires exiting the board from the bottom going downward go to a second “external” PS103J2 thermistor which can be used to monitor temperature elsewhere – typically the outside environmental temperature. These thermistors are sampled with a 12-bit ADC, so can measure temperatures from -80 to +150 degrees C to within a tenth of a degree. There are also two sets of 3-pin headers to control servos, one on the left edge at the top and a second just to the right of the voltage regulator.

The breakout boards, all mounted on plug-in headers so they can be removed then screwed down to stand-offs so they do not pop out on hard-landing impacts, are as follows. In the lower right is an XB3-24Z8PT-J XBee3 radio module, for short-range communication with other payloads (such as other PTERODACTYLS, a venting device, and/or an XBee-enabled Iridium tracking unit) on the stack. Directly above that is a (red) LSM9051 9DOF IMU. (Additional details about the on-board (AKA “basic”) sensors – IMU, pressure, gps, temperature – are given in the following section.) In the upper right is a (blue) GY-63 MS5611 pressure sensor breakout board. In the upper left is a (white) u-blox NEO-M8N gps module which must be set to airborne mode for stratospheric use. In the lower left is a (red/black) tiny LCD-14532 OLED screen, 64 pixels wide and 48 pixels high, on which messages and sensor data can be displayed. Typical OLED screen output showing gps, dual temperature, and pressure sensor values appears in Fig. 5. The thermistors are analog devices, the pressure sensor, IMU, and OLED are I2C devices, and the gps and XBee modules use serial RX/TX ports.



Fig. 5: Typical OLED output from a PTERODACTYL, showing gps, two temperatures, and pressure.

Around the perimeter of the PTERODACTYL v2 pcb are headers to give user access to some of, though admittedly not all of, the other pins on the Teensy. Explicitly labeled are RX1-TX1 (top edge) and RX2-TX2 (left edge) for serial communication, as well as SDA-SCL (right edge), and SDA(1)-SCL1 (bottom edge) for I2C devices. On the right side in the photo in Fig. 1 an I2C sensor (not visible) is wired the SDA-SCL pins, plus nearby 3.3 volt and GND.

3. The “Basic” Sensor Suite

As mentioned earlier, the PTERODACTYL has a set of built-in sensors to collect and record basic data. The built-in sensors are an MS5611 pressure sensor (using which altitude can be estimated), a 9DOF IMU, a u-blox NEO-M8N gps, plus an “internal” (on-pcb) thermistor and a terminal block for an “external” thermistor. The data from these sensors provides information about how basic environmental conditions changed throughout a flight, and where the PTERODACTYL was located when the data was collected. Data is logged in .csv format onto a micro-SD card, typically about once a second, and can be transmitted by XBee radio as well. Plots of some of the data from a PTERODACTYL unit from a stratospheric balloon flight are shown in Fig. 6 and Fig. 7.

The pressure sensor on board the PTERODACTYL can be used as an altimeter, by applying a manufacturer’s calibration curve to the pressure data to interpret it as an altitude (estimate). In the plots mentioned above, the gps altitude dropped out periodically during ascent and the gps froze entirely early in descent, so the pressure-based altitude estimates were particularly valuable. Even if the gps works well, it is instructive to compare altitudes estimates based on pressure with “real” altitude reported by the gps. Depending on the pressure sensor, there is typically a critical altitude (i.e. a low pressure threshold) beyond which the pressure-based altitude estimates become less and less accurate. The MS5611 pressure sensor on the PTERODACTYL does a fairly good job in estimating altitude up to 100,000 ft ASL, as shown in the plots, but starts to struggle at even-higher altitudes that can be reached by certain low-payload-weight weather balloon flights.

The external temperature shown in the plots mentioned above is somewhat noisy, but clearly shows the temperature dropping to about -50° F at the tropopause both on ascent and on descent, with the environmental temperature increasing with increasing altitude in the stratosphere until burst. The internal temperature of the box stayed quite warm due to the PTERODACTYL being mounted inside an insulating payload box that was painted black which also contained some

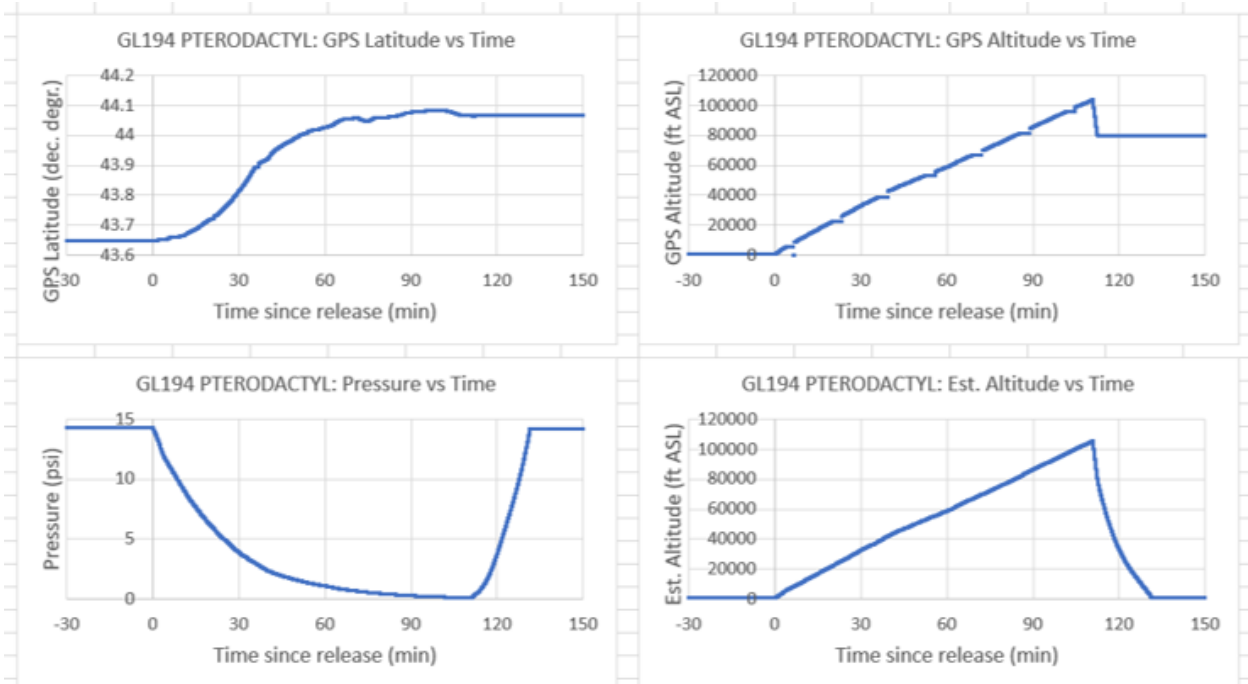


Fig. 6: Sample GPS and pressure flight data. The gps failed early in the descent, and also had periodic altitude drop-outs on ascent (an issue we are trying to fix), but the pressure-based altitude estimates were quite accurate all the way to burst just above 100,000 ft ASL.

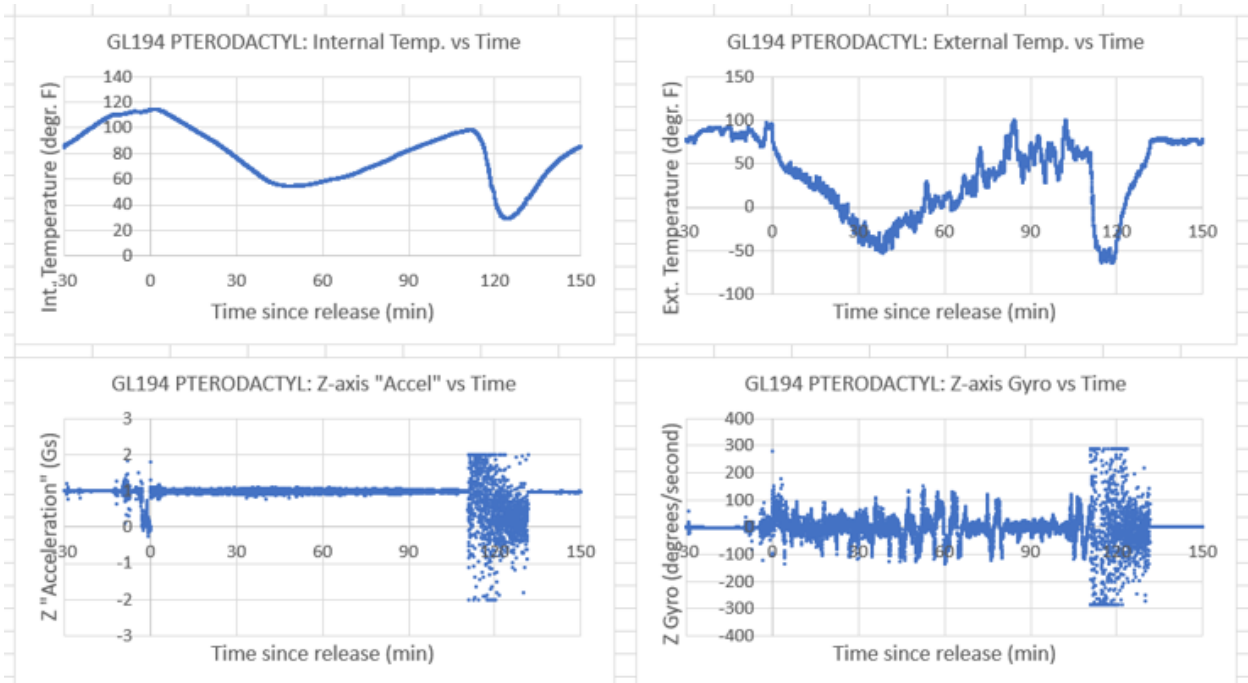


Fig. 7: Additional “basic” data from the same flight as Fig. 6. The internal temperature stayed above freezing for nearly the entire flight, while the external temperature dipped below -50° F. “Accel” about the z-axis (the vertical axis) shows a smooth ascent, somewhat bumpy parachute descent, and an upright landing. Rotation about the z-axis was modest, except for a quiet stint late in the ascent, but quite wild (sometimes saturating the gyro) on descent.

chemical hand-warmers. Dips in the internal temperature occur as the payload passed through the tropopause, both on ascent and on descent, but these are slightly delayed from the external temperature record due to the time lag associated with the box insulation.

The 9DOF IMU on the PTERODACTYL includes a 3-axis accelerometer (which actually measures force, not acceleration, so it might be better-described as a 3-axis force sensor or perhaps a 3-axis tilt sensor), a 3-axis gyroscope, and a 3-axis magnetometer. The tilt and rotation data from this sensor gives a good idea of how the balloon stack moved before, during, and after the flight, as opposed to characterizing the environment through which it moved. Accelerometer data, in particular, is useful to time-tag the release, the burst, the landing, and also the general turbulence (always present on parachute descent (AKA “post-burst chaos”); sometimes present on ascent as well). The sample plots show accelerometer and gyroscope data, both with respect to the z-axis (the vertical axis). Burst and landing are easy to identify, though not so much the release because of pre-release handling of payloads.

4. Additional Applications That Have Used PTERODACTYL

Since its introduction in 2020, we have used the PTERODACTYL to log data from a variety of additional sensors, to test sensors for stratospheric applications and to collect data for unique projects. For example, we have used a PTERODACTYL to power and log data from an Aware Electronics RM60 Geiger counter to measure cosmic radiation. We have also logged data from a variety of I2C sensors from Sparkfun (often “QWIIC” sensors; mostly light detectors) and from Adafruit, admittedly with mixed results. We have also used a PTERODACTYL to call for mock-PPOD deployment events of mock-CubeSats [2]. Indeed, one of the pre-wired switch settings is explicitly for mock-PPOD applications.

On the radio front, we have also used a PTERODACTYL to command a vent device attached to the balloon carrying the stack [3]. We have added an RFD900 radio to a PTERODACTYL and used it to transmit data to a ground station while in flight. We also added a second XBee3 module to a PTERODACTYL for use as a radio-relay, to send data to a Stratostar Satcom unit. In that case, the original XBee3 collected data by radio telemetry from other PTERODACTYL units on the same stack, the data was combined and reformatted, then the second XBee3 radio module relayed it on to a SatCom unit which transmitted it to the Iridium satellite network so that we could pick it up on the ground in (near-)real time.

5. PTERODACTYL “v3” with Teensy 4.1 (Blue PCB)

In the summer of 2022, we revisited the PTEROCATYL pcb design due to difficulty in sourcing Teensy 3.5 microcontrollers and certain other parts, including XBee3 radio modules. The new version, called “v3” or “blue” since it is printed on a blue pcb, uses the Teensy 4.1 microcontroller which is faster than the Teensy 3.5 but has a slightly different pin-out and on which digital pins are not tolerant of voltages above 3.3 volts. In this section “PTERODACTYL” always refers to the “v3” (Teensy 4.1) variant. For posted documents about the PTERODACTYL “v3” see Ref. 4.

When designing the new PTERODACTYL, one minor error on the earlier pcb was fixed: the power switch on the earlier board did not turn off power to the voltage regulator. Extra headers to plug in the gps were added, allowing the board to be populated with either a u-blox NEO-M8N or a u-blox NEO-M9N gps module. The power pins were gathered together along the top edge, as oriented in Fig. 8 and Fig. 9 (schematic in Fig. 10), and the data pins were collected together along the bottom edge of the pcb, with a full set of parallel connections for (plug-in) female headers and

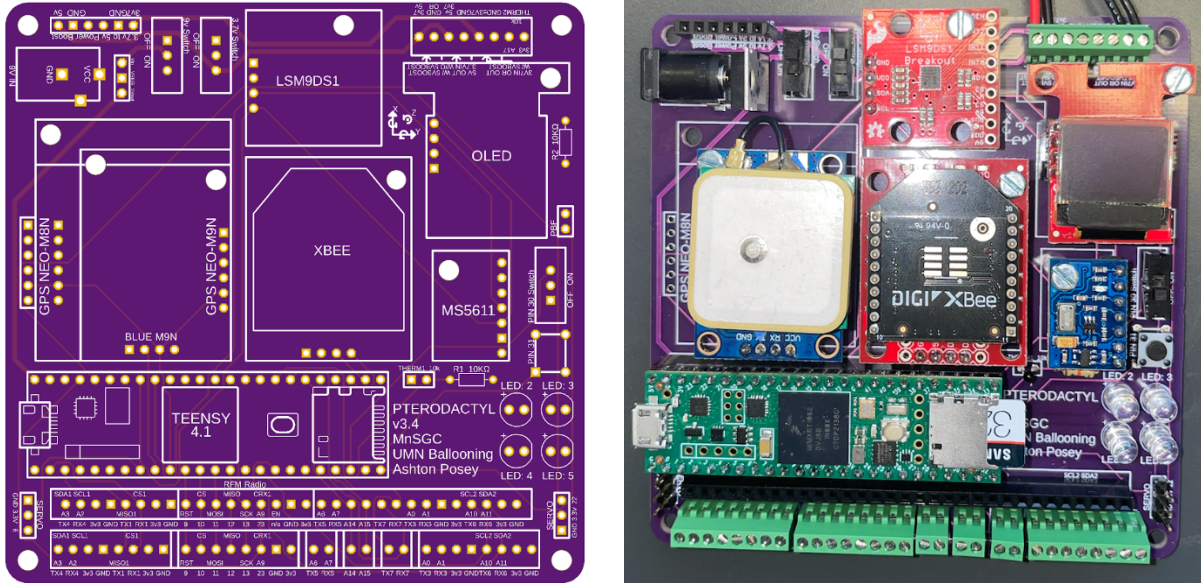


Fig. 8: Bare and populated pcb for the Teensy 4.1 version of the PTERODACTYL flight computer.

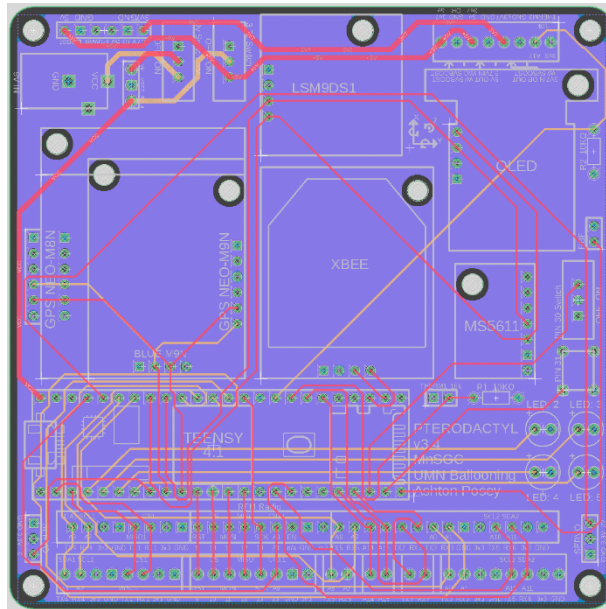


Fig. 9: Wiring layout, drawn with Eagle software, for the custom PTERODACTYL pcb (Teensy 4.1 version).

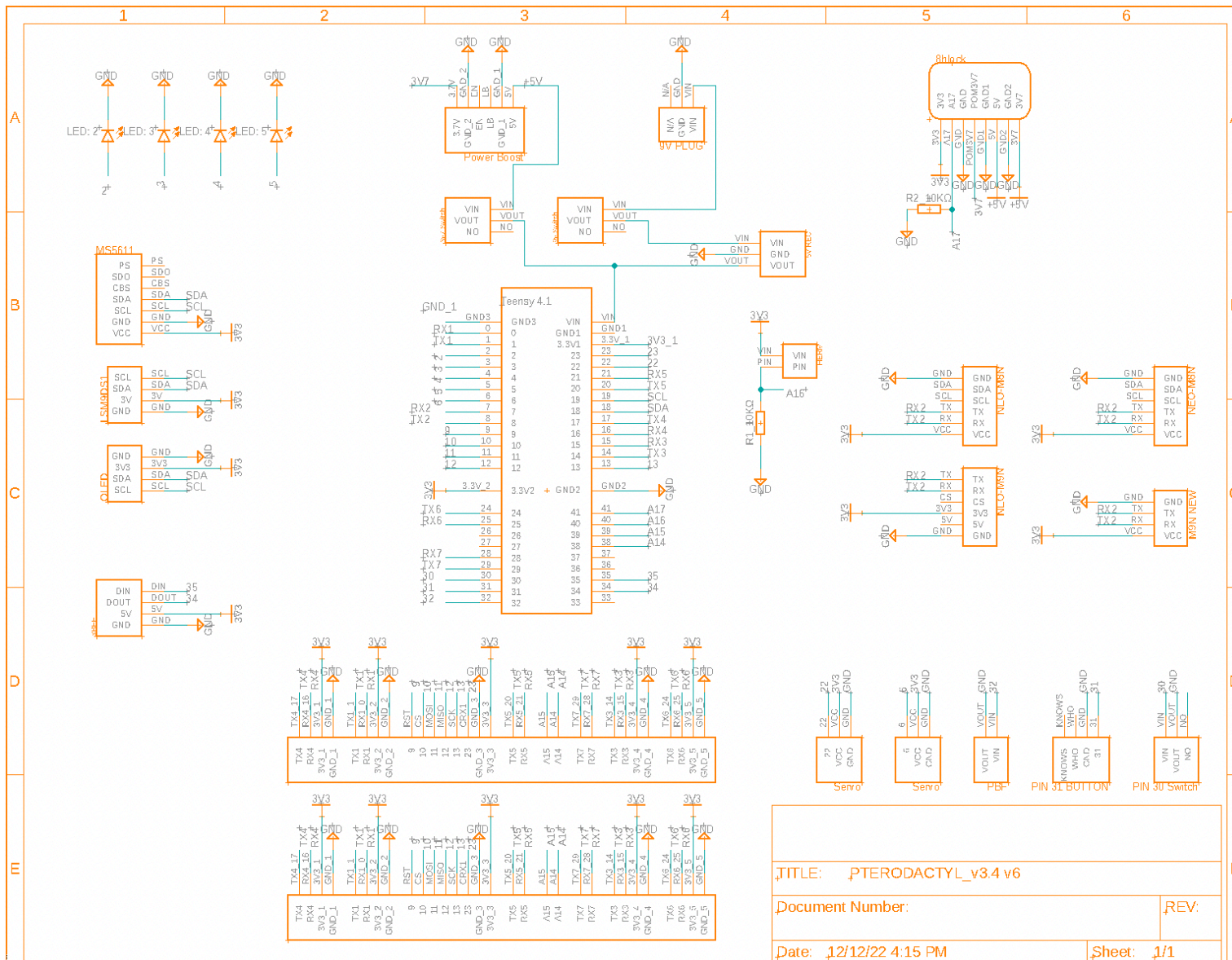


Fig. 10: Schematic of PTERODACTYL (Teensy 4.1 version).

(screw-in) terminal blocks to give the user more options. The number of available serial ports increased from 5 to 8. An explicit header was made for an RFM69 radio from Adafruit – our planned backup, if XBee3 radio modules cannot be purchased.

The “basic” sensors and breakout boards remained the same, though the physical layout on the pcb was significantly changed, as was the flight code. For example, code was written to run the gps module (either the u-blox NEO-M8N or the u-blox NEO-M9N) in “precision” mode using ECEF (Earth Centered, Earth Fixed) coordinates, which have higher resolution than the default lat/long/alt mode. A new version of the “Power Plane” was also developed, to accompany the new PTERODACTYL – see next section for details. On the new PTERODACTYL pcb the number of switches and shorting plugs was reduced – they were rarely being used – and power options were expanded to make it easier to power the PTERODACTYL using a 3.7 volt battery, if desired.

6. The “Power Plane” with Additional Functionality

The “Power Plane” custom pcb has the same form factor as the PTERODACTYL and can be mounted, using standoffs, above or below the main board - we prefer below (see Fig. 11, with standoffs reaching “up” for mounting a PTERODACTYL) so that the Power Plane does not prevent the user from reading the OLED screen on the PTERODACTYL and does not interfere with the up-looking antennas on the gps and XBee modules. The Power Plane has places to mount batteries (two 9 volt batteries, wired in parallel, or else one 3.7 volt battery pack), 3.3 volt relays that can be used to turn various devices on and off (such as a heater or a siren) that can be powered with less than 30 volts at 2 amps (60 watts), plus a header for additional I2C sensors. When 9 volt batteries are used, the voltage is again stepped down to 5 volts using a linear voltage regulator. A schematic of the Power Plane appears in Fig. 12.

The relays on the Power Plane are a useful addition to the PTERODACTYL system because they allow devices to be turned on and off using the microcontroller's 3.3 volt logic, instead of needing to throw a physical switch either manually or with a servo. The relays work by magnetizing one side of the relay to “set” it in the ON position, then magnetizing the other side in order to deactivate the “set” side and activate the “reset” side instead. The relays on the Power Plane have two data lines and two ground lines to control the relay, plus 3 pins on each side of the relay that can set the direction of the current on their side but have no continuity with the other side. Thus, if two or three devices need to be turned on and off together, but have separate batteries, a single relay could be used. We have used relays to turn off a siren after a payload has left the ground (and turn it back on for landing/recovery) and also to activate a mesh heater to warm the batteries powering the Power Plane and/or PTERODACTYL itself. Heaters are especially important near the tropopause where the external temperature is so low it could potentially stop batteries from working. We sometimes use chemical hand warmers to try to heat components, but electrical heaters can be more targeted and are less wasteful because they can be programmed to only turn on when a temperature sensor indicates they are actually needed.

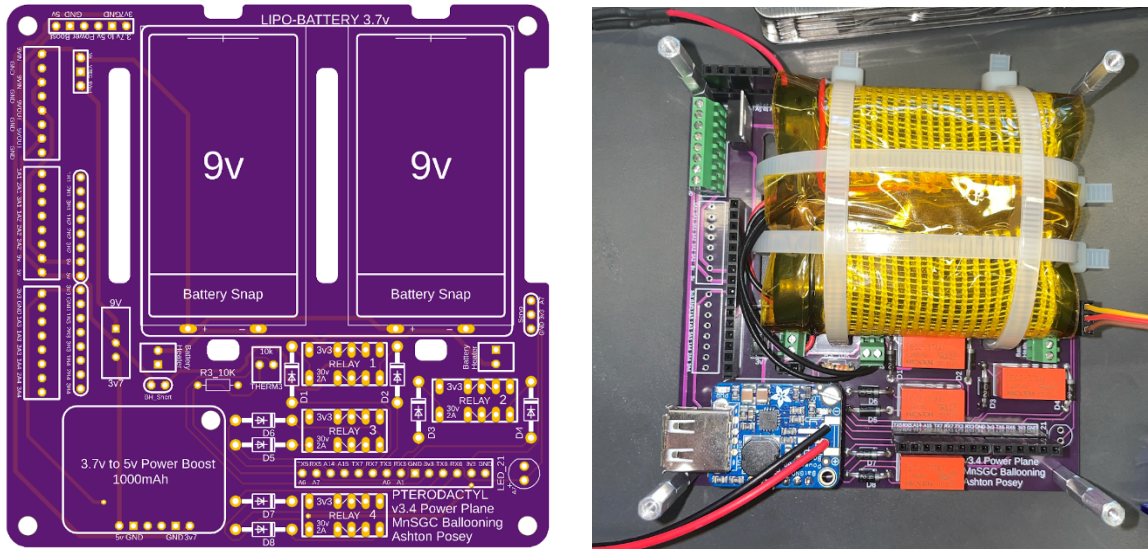


Fig. 11: Bare and populated “Power Plane” PCB shield for the PTERODACTYL.

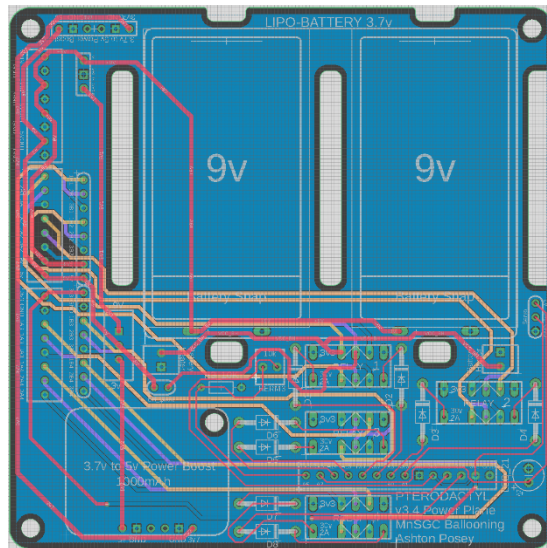


Fig. 11: Wiring layout, drawn with Eagle software, for the custom Power Plane pcb.

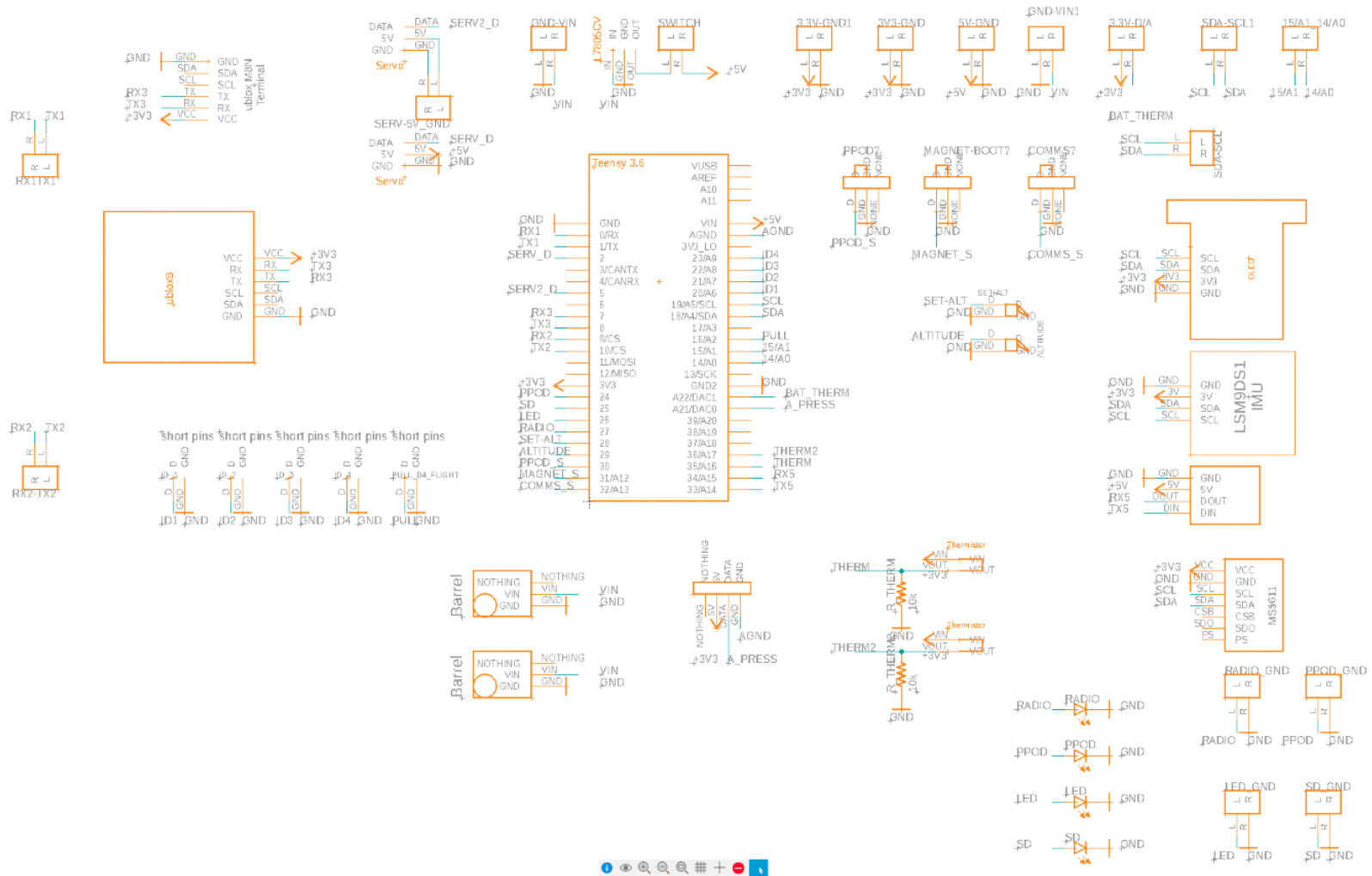


Fig. 12: Schematic of the Power Plane.

7. Potential NEBP (Eclipse Ballooning) Applications

The PTERODACTYL has been selected as one of several flight computers to be used by teams participating in the “engineering” side (AKA the “video-streaming” side) of the National Eclipse Ballooning Program (NEBP). Modifying the flight code to capture higher-precision gps coordinates was in response to a direct request of the NEBP. Hopefully this will allow video-streaming “heavy” payload stacks to contribute to the data being collected by the “Atmospheric Sciences” side (AKA the “radiosonde/meteorological gravity wave detection” side) of the NEBP.

The communication functionality of the PTERODACTYL, whether provided by XBee radio modules or something else which is actually in stock, will also be valuable on NEBP flights. For example, the NEBP “video-streaming” balloon flights will use XBee-enabled Iridium modems for tracking balloons and for pointing ground stations. The Iridium system allows for uplink of commands to the in-flight modem, which could then be relayed to a PTERODACTYL which might have radio control over a vent and/or a termination/cutter device. A PTERODACTYL could also potentially relay status reports from payloads to the ground through an Iridium/XBee radio link.

8. Summary

The PTERODACTYL flight computer has been a valuable device for a wide range of ballooning projects. It weighs only about 5 oz (plus battery(ies)) and the parts cost is less than \$300 (“basic” sensors only). It requires only basic soldering skills to assemble - no surface-mount soldering - and the Teensy microcontroller can be programmed using the Arduino IDE. The basic sensor suite is useful for introductory applications, but the ability to add additional sensors and the radio telemetry features make it a useful device for more-advanced projects as well. Its 10 cm x 10 cm form factor will be familiar to CubeSat developers. We look forward to assisting with nationwide adoption of the PTERODACTYL by eclipse ballooning teams in the NEBP.

Acknowledgments

The authors gratefully acknowledge support from the NEBP, for which the U of MN - Twin Cities ballooning team is serving as a “pod lead,” and from the MN Space Grant Consortium.

References

- 1 Van Gerpen, A. & Flaten, J., (2022) “Payload To Enable Recording Of Data And Communication Telemetry Y(While) Lofted (PTERODACTYL), with Stratospheric Ballooning Applications”, *Academic High Altitude Conference 2020*(1). doi: <https://doi.org/10.31274/ahac.11646>
- 2 Wehling, P. & Van Gerpen, A. & Flaten, J., (2022) “Flying “Mock” CubeSats on Stratospheric Balloon Flights: An Update about “Mock P-PODs” and the 3U “MOC-SOC””, *Academic High Altitude Conference 2020*(1). doi: <https://doi.org/10.31274/ahac.11647>
- 3 Larson, S., Cook, C., Wallo, S., Thompson-Jewell, E., Posey, A., Meyer, J., and Flaten, J. “Development of an Adaptable Vent for Latex Weather Ballooning”, *Academic High Altitude Conference 2022* (to be published in conference proceedings).
- 4 Link to PTERODACTYL v3.4 pcb files and code <https://drive.google.com/drive/folders/1AHkEVTcYGjr29ESmD6pJE1bOII95peQj?usp=sharing>