Project Caelus is a non-profit rocketry organization I helped start in high school. The team aims to be the first high school rocketry team to develop a liquid-fueled rocket engine and launch it to the edge of space (100km above the Earth’s surface). Over my three-year tenure with the team, I helped develop the engine test stand, a testing platform that allowed us to validate our engine design, flight software, and electrical systems.

In particular, as the head of the Avionics sub-team, I led a team of four to develop an avionics system that could support the processing needs of our flight software; make accurate pressure, temperature, and fuel measurements; and automatically toggle various solenoid valves to control the fuel/oxidizer outputs to the engine. The system also needed to be able to allow human operators to manually override the flight software and abort from a safe distance.

**Design Overview**

At the heart of the system is a two-layer copper PCB that hosts a Teensy 3.6 flight computer running C++ flight software. The system can simultaneously:

* Take measurements from:
  + 12 Pressure Transducers
  + 5 Thermocouples (temperature sensors)
  + 3 Load Cells (force/weight sensors)
* Actuate up to 12 solenoid valves
* Transmit all sensor measurements/valve states/additional telemetry to ground station
* Provide hardware override for ground station operators in need of abort

With the onboard batteries, the system could be powered continuously for up to three hours, which was necessary for long-duration tests. All parts were conformal coated, protecting them from dust, moisture, and temperature extremes.

**Design Process**

After receiving the design requirements and constraints from the Propulsion team, the Avionics team got to work designing a schematic for the system in KiCAD’s Schematic Designer.

Pressure Transducers (PTs)

Each PT provides three wires: Red (5V+), Black (GND), and Green (Signal). The sensor outputs a 5V signal that needs to be attenuated to a 3.3V signal for our Teensy 3.6 microcontroller. To achieve this, every signal input is routed through a voltage divider that cuts the signal by a third. Due to high power losses across the resistors, future iterations will utilize Inverting Op-Amps with a fractional gain to achieve the attenuated signal.

Thermocouples

Each K-type, ungrounded thermocouple requires an amplifier to make the millivolt output readable by the Teensy 3.6 flight computer. For this, we chose the Adafruit MAX31856 amplifier due to its compatibility, reliability, and affordability. The MAX31856 communicates with the FC via the SPI serial protocol.

Load Cells

Load cells are essentially electronic strain gauges, so they can be used to measure the weight of our tanks and the thrust generated by our engine. Similar to the thermocouples, each load cell outputs a millivolt signal which needs to be amplified for the FC. For this, we use the SparkFun HX711 amplifier which communicates with the FC via serial communication.

Solenoid Actuation

Solenoid valves are relatively simple devices – internally, they contain a coil of wire attached to a plunger that opens and closes the valve when a current is applied. The Teensy 3.6 cannot supply enough current to actuate these valves; thus, the FC controlled Arduino Relays, which supplied power from our 6S Lipos to open/close the valves. In future iteratios, the Arduino Relays will be replaced by power transistors which consume less power during operation.

However, to allow for groundstation override capabilities via the Launchbox (explained later), each relay needs to be connected to both the FC and the groundstation. To achieve this, every relay signal line was attached to a DG419 IC (essentially an electronic switch) that allows the relay to be controlled by both input sources. When the Launchbox is enabled, the enable pin for each DG419 is toggled HIGH, thereby connecting each relay to the groundstation and allowing for manual override.

Data Transmission

All sensor measurements, valve states, and flight software variables are transmitted to the ground station via an Xbee radio. Data packets are sent 10 times a second, allowing for ground station operators to get a comprehensive view of the state of the entire system at any moment.

Power Distribution Board (PDB)

As mentioned in the Solenoid Actuation section, every solenoid valve needs to be powered via an external battery source (in our case, a 6S lipo). The Power Distribution Board (PDB) provides multiple 24V and 12V terminals to power the various solenoids on the test stand. The board is powered via a 6S Lipo (24V battery) that runs through an external step-down converter to power the 12V terminals. Future iterations of the board will include built-in XT-60 connectors instead of solder pads to allow for better wiring organization.

After developing the schematic, laying out the footprints in KiCAD’s Layout Editor, drawing the copper traces, and generating the gerber files, we sent our plans out to be manufactured by JLCPCB.

**Testing**