Since July 2023, I have been the Ground Systems lead for Rocket Project’s flagship liquid team. The goal is to launch a bipropellant rocket, fueled by a blend of liquid oxygen and ethanol, over 30,000 feet. I’ve been responsible for designing, manufacturing, and testing the electrical systems responsible for propulsion controls, remote fill, and data collection.

My system integrates three main subsystems interconnected through a Raspberry Pi 4. The switchbox system is housed in the bunker at our test site, while the Raspberry Pi, controls, and DAQ systems are located at the launch pad. Using USB connections, the Pi links to the controls and DAQ systems. The Pi also acts as a local server, facilitating remote communication for our team in the bunker. In real-time, the Pi uploads data from DAQ to a Grafana dashboard, enabling rapid assessment of the performance of propulsion systems during tests.

The switchbox, as the name implies, is a box of switches that sends commands that activate and deactivate solenoid valves at the pad. My team developed a wireless switchbox system that uses an ESP32 to transmit commands over Wi-Fi to the Raspberry Pi. We started with breadboards to validate component selections and to write the code. Once the design was thoroughly tested, my team designed a simple PCB to house all the components. We also made a wired switchbox system that sends commands via RS-485 directly to the control system. I decided to have this redundant system as a failsafe in case the Pi failed.

The control system, triggered by commands from the switchbox, is designed for precise control over solenoids, either supplying or cutting off 24V as needed. I implemented this functionality using an Arduino that controls 16 electromechanical relays through a 16-channel IO expander. Each relay, connected to a solenoid, establishes continuity between the common and normally open terminals when signaled by a low input from the IO expander, delivering 24V to the solenoid. I employed diode safeguards against back EMF, ensuring circuit integrity. Rigorous testing on breadboards, akin to the switchbox, validated component performance and wiring accuracy. With a verified schematic, I developed a PCB housing the Arduino, IO expander, diodes, and a MAX485 module for the wired switchbox setup. The PCB also accommodates two 8-channel relays that directly plug into the PCB. Given the critical safety implications, I conducted thorough tests to confirm the solenoids' reliable actuation and deactivation. Additionally, I designed an automatic abort feature, initiated by a 30-second timer in the absence of switchbox communication, to safe state the solenoids if connectivity is not restored within the designated timeframe.

The DAQ system logs critical parameters such as pressure, engine and tank temperatures, tank fill levels, and thrust. Currently, the system accommodates 10 pressure transducers, 5 thermocouples, and 5 load cells. I designed specialized PCBs to handle analog signals from thermocouples and load cells, amplifying and processing them through ADCs, with an ESP32 managing thermocouple data and an Arduino handling load cell data. I mounted a 10-channel ADC on the Pi to process pressure data to conserve space. I designed another PCB that employs optocouplers to convert 24V solenoid signals to 5V signals. They can then be processed by an Arduino, enabling remote confirmation of the solenoid states without on-site personnel. The Pi efficiently processes and simultaneously stores and streams all data to a Grafana GUI.

I am currently working on a new loadcell system that optimizes the Wheatstone bridge circuit to have a several times faster sample rate than the current system. Additionally, my team is working to develop capacitive fill sensors for the tanks. Our current system utilizes loadcells to approximate the fill level. However, this has proved inaccurate. Measuring the capacitance change as the tanks are filled should provide more accurate data. Stay tuned for updates!