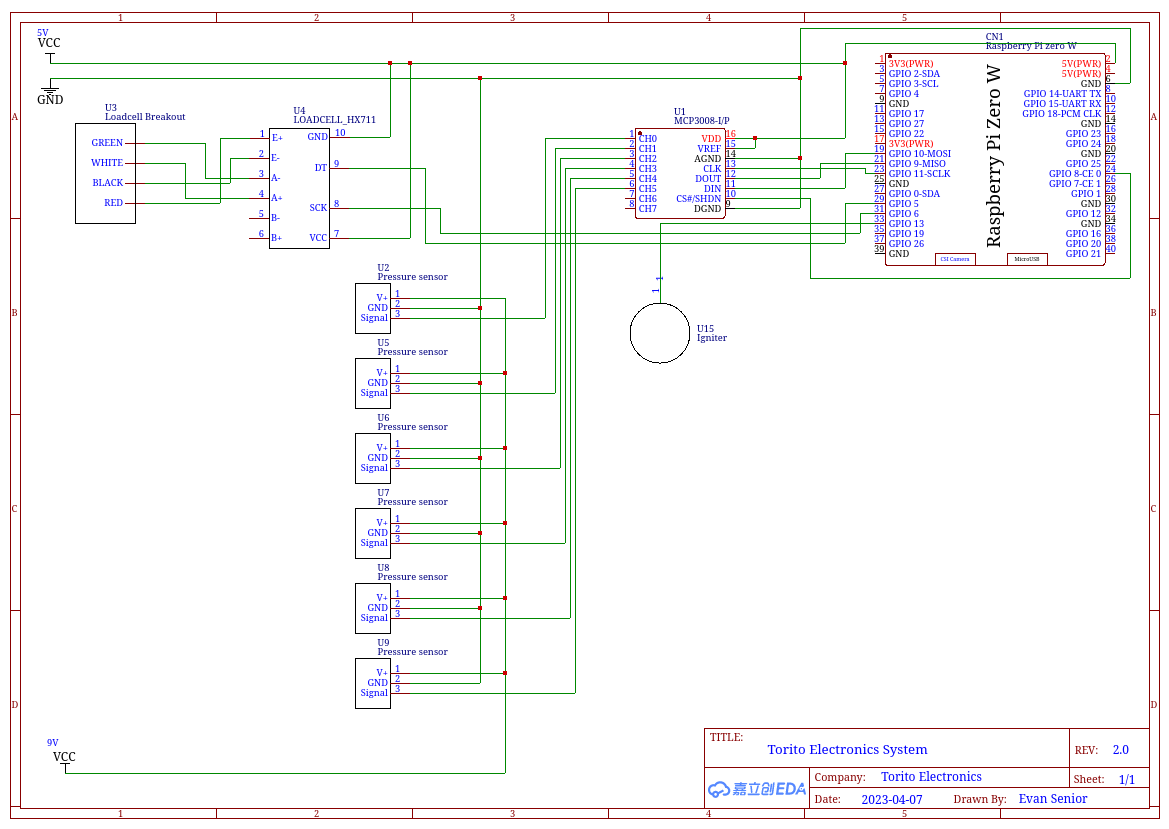
The way that the sensors will be setup is that they will be interfacing with a raspberry pi zero w. Since the raspberry pi does not have analog to digital pins. This will be counteracted with an analog to digital converter, the MCP3008 chip on Figure 2. This chip is able to input up to 8 different analog to digital pins allowing for more sensors if needed. The way that the design is setup is so that there will be 6 pressure sensors. The temperature sensors, thermocouples, will be able to be read from a raspberry pi plate, the MCC 134, and the plate will sit on top of the raspberry pi. The plate is not featured on Figure 2 because it could not be located in the EasyEDA Library. The plate is able to intake up to 4 thermocouples, 8 wires total and if needed multiple plates can be stacked on top of each other for more input. All the sensor's desired locations are featured in Figure 1. There will be 6 pressure sensors, one on each tank, two at each tanks’ outlet, and the final two at each of the injector inlets. The pressure sensors will be pressure transducers with wire leads and a 3 wire connection. The temperature sensors, thermocouple, will have locations of the two tanks, one on the face of the combustion chamber on the outside, and the last one on the outside of the nozzle. The tank thermocouples can consist of thermocouple probes to read the temperature of each tank. The load cell will be connected to an amplifier as shown in figure 2, and record the force generated by the rocket. The voltage will either have 2 sources, one from the raspberry pi and the other from a 9V battery.

Another solution to this is to have one larger battery of greater voltage and regulate it using a regulator to distribute power of different voltages to both the raspberry pi and the pressure sensors.

The code for the raspberry pi is a combination of included libraries found on the internet as well as original code to make it work together with a python program to showcase all of the sensor data in a gui format. The gui is updated in half a second intervals so that the most accurate readings from the sensors that are being fed into the raspberry pi from the various sources. The interval of reading sensors can be modified according to the desired interval of time in seconds. The method of communication for the raspberry pi is through ssh, where a computer can connect to the raspberry and it will have the code already on it for reading the sensors and a gui will be displayed to the connected computer, all the connected computer will need to do is run the python script. The data will be displayed on the gui as well be saved in a csv file, a comma separated list, and will be easily accessible through an excel program.



**Figure 2: Schematic of the Raspberry Pi Zero W Circuit**

The location of all the electrical components will be located on the test stand itself. The raspberry pi will be securely connected in front of the load cell where it will not be damaged if the nozzle and combustion chamber become loose. It will also be in a secure location as to if something does go wrong in a bad case, the data from the raspberry pi will be collected and stored not just on the raspberry pi but also on the host machine connected to the raspberry pi every refresh interval, half a second. The code also saves the data file in a csv format every interval so if the sd card used for the pi is not damaged but the pi itself is damaged the data is still retrievable.