LoRa Water Quality Management System
PCB Design

Team Members:

Cole Hadley, Patrico Benitez Palermo, and Matthew Sharp

Technical Advisor: Dr. Aldo Morales

Date: 11/21/2025

PCB Design Overview

Though PCBs were designed for both the sensor node and central gateway of the LoRa Water Quality Management System, the PCB which required the majority of the engineering labor was the sensor node PCB. This part of the system is responsible for interfacing directly with the turbidity, temperature, and pH sensors which will inform the decision-making logic at the central gateway. The sensor node PCB integrates sensor interface circuitry, the signal conditioner subsystem, a microcontroller, the LoRa transceiver module, USB, UART, and SWD interfaces, along with indicator LEDs and a RF signal path.

The PCB was entirely designed using KiCAD, a freely available PCB design software. In addition to the features available within KiCAD the RF-Tools plugin was used to manage critical high-speed signal paths, as well as Saturn PCB Toolkit to determine PCB trace geometries for impedance-controlled traces. Though most of the traces carry low-speed analog and digital signals, there were a few special considerations that needed to be made when developing the final design. First, the design involved a RF signal trace connecting the LoRa transceiver module and the external antenna, as well as USB 2.0 data traces. To maintain signal integrity within the PCB and prevent potentially damaging signal reflections, the RF signal trace had to be impedance controlled to 50Ω, while the USB 2.0 traces needed to maintain a differential impedance of 90Ω, $\pm$15%. Next, the power supply into the sensor node needed to accept a wide input voltage range, manage heat dissipation effectively, and accommodate large switching currents through its inductor. Another key area of special concern is power distribution, as the PCB must accommodate four distinct voltage rails throughout the design, while maintaining low impedance and efficient routing. The microcontroller layout was another area of concern within the design, as the chosen RP2350 microcontroller requires a substantial decoupling network, introduces a new power rail, and provides all of the external interfaces on the board. Finally, the PCB must minimize the propagation of high-frequency noise throughout the rest of the traces and outside of the PCB to improve signal integrity and minimize unwanted external noise.

The PCB can be divided into four main sections: power input and regulation, microcontroller layout, analog front-end, and RF interface. To minimize the effects of high-frequency noise on critical traces, the RF interface and switching regulators were isolated from the microcontroller and analog front-end. The main subsystem that this PCB was designed to accommodate is the Signal Conditioner, which is the subsystem that was designed “from scratch” for the purpose of this project. The purpose of the signal conditioner subsystem is to provide a unified analog front-end which will be shared among all the sensors used with the system. It applies DC offsets and gain to the incoming signal, making them compatible with an analog-to-digital converter (ADC). The value read by the ADC is read by the microcontroller, which formats the incoming data as real-world

measurements, and transmits the telemetry to the central gateway using LoRa. The gain and DC offsets of the signal conditioner are controlled using digital potentiometers, making the signal acquisition process fully software-controlled and autonomous. Outside of managing the parameters of the signal conditioner and the LoRa interface, the microcontroller also manages the power consumption of the PCB, controlling the 5V rail by enabling and disabling the power regulator IC, disconnecting the sensors and minimizing power draw while the system is idle.

PCB Design Steps

Before the PCB routing was initiated, the schematic design was first finalized using KiCAD for schematic capture and LTSpice for analog simulation. Full schematic captures of the design have been attached to the end of this report. The RP2350 schematic largely follows Raspberry Pi's reference design guide, with some modifications made on the USB boot control and reset to make operation more reliable. The major component selections for the sensor node were made as follows, with corresponding justification:

Power Regulation

- Texas Instruments TPS63000 Buck-Boost Regulator IC: Wide input voltage range (1.8-5.5V), adjustable output from 1.2 to 5.5V, 90% typical operating efficiency, single inductor for buck and boost mode operation, robust reference design and documentation. Schematic follows the reference design with an oversized output capacitor.
- Texas Instruments LM2660 Switching Capacitor: 100mA output current, straightforward design, Vout = -Vin, compatible with 5V input.

Microcontroller

- Raspberry Pi RP2350: Robust C/C++ Software-Development Kit, two I^2C and SPI peripherals integrated on-chip, many GPIOs, high clock speed, robust power management capabilities.
- Texas Instruments LSF0102 Logic Level Translator: Straightforward Design, 3.3V-5V compatibility.
- Macronix MXL25L3233F NOR Flash: SPI Interface, Non-volatile memory, 32Mbit
- Winbond W25Q128J NOR Flash (MCU Program Memory): QSPI Interface, High Datarate, 128Mbit suitable for most firmware builds, supported by MCU bootloader

Analog Front-End

- Texas Instruments TMUX1309 MUX: Differential, Analog Pass-through capability

- Texas Instruments TLV9144 Op-Amp: Low Power consumption, rail-to-rail input and output, high CMRR, high input impedance, quad package
- Microchip MCP4651 Digital Potentiometer: Two Potentiometers on-chip, I^2C control
- Microchip MCP3425 ADC: Sigma-Delta Design, 16-bit precision, I^2C Interface
- Texas Instruments TL082 Op-Amp: High Input Impedance, Dual Package

RF Interface

- SX1262 LoRa Modulator: Industry-Standard IC, robust documentation, manufacturer-provided SPI lookup table directly compatible with C-based firmware
    - SX1262 implemented using Waveshare Core1262-HF LoRa Module, selected as it simplified the design process and has integrated EMI shielding.

With the component selections made, the component footprints were downloaded from the respective component manufacturers where available, and standard IC package footprints were used when models were not directly available. From there, the components were segregated based on to which PCB category they belonged – the RF interface and power supply were located towards the corners of the board to minimize noise, while the microcontroller was located centrally to maximize access to all digital signals. The analog frontend was placed at the bottom of the board for proximity to the sensor connectors. All other connectors were located such that they were close to the nets they would interface with. Before traces were routed, the board stackup and net classes were defined, producing a 4-layer board with impedance control on the USB traces and RF signal path. The USB geometry was defined as a differential pair with 0.2mm track width and 0.14mm separation, giving a 90.049Ω differential impedance according to Saturn PCB toolkit. The RF signal path was defined as a coplanar waveguide with 0.35mm track width and 0.15mm clearance, giving a 50.13Ω characteristic impedance. To maintain the topology needed for the differential traces and coplanar waveguide, the stackup of the board was defined to be signal traces, ground, power rails, signal traces and ground, from top to bottom.

The first section that was routed was the power regulator, shown below. This section's key concerns included thermal management and managing potentially high currents. Stitching vias were used on the power input to increase surface area for heat dissipation, and on power outputs to provide a low impedance path to the power planes. Traces to the inductor were maximized in available width, supporting up to 1A of current through the inductor, well above the calculated switching current.

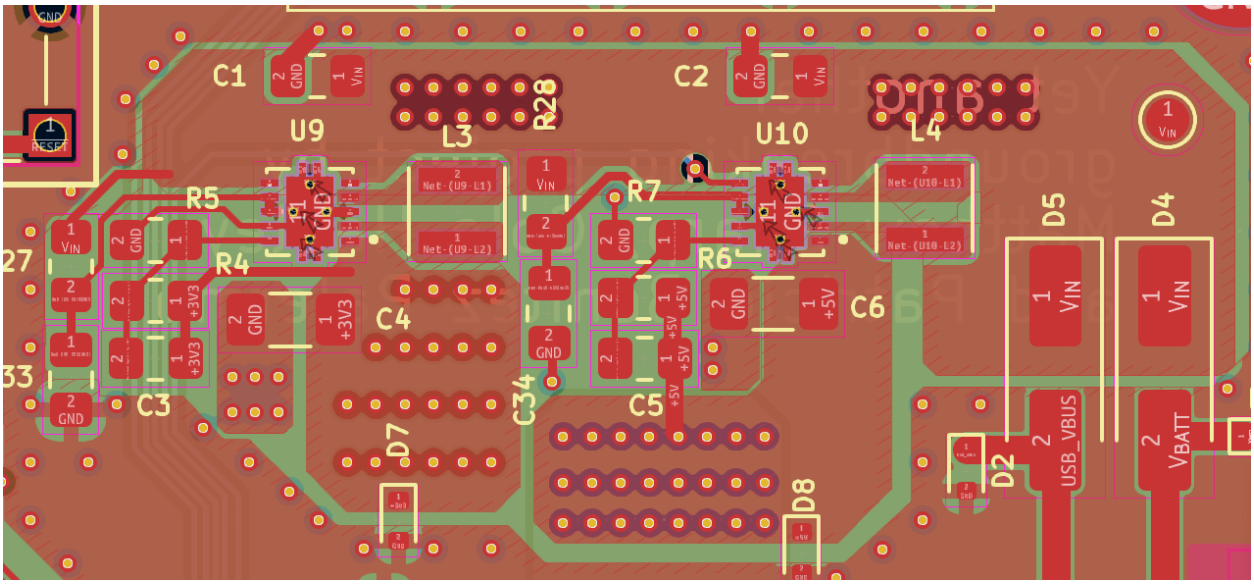


Figure 1: Power Regulation Section of PCB, as Produced in KiCAD

The next section to be routed was the microcontroller, which relied heavily on the manufacturer's reference design, however it needed to be adapted for the 4-layer board stackup and larger 0805 surface-mounted components. The RP2350 is shown as U8 below, with USB traces run directly to the USB port through 27Ω termination resistors. The internal power and ground planes have been hidden in the viewport shown to show the 1.1V copper pour used to supply the internal logic core of the microcontroller. This rail is produced by an internally-controlled buck regulator inside the RP2350 MCU, with external inductor and capacitors required for operation, shown as C20, C21, and L2 below. Key attention needed to be paid to keep these components as close to the footprint of the chip as possible while maximizing ampacity.

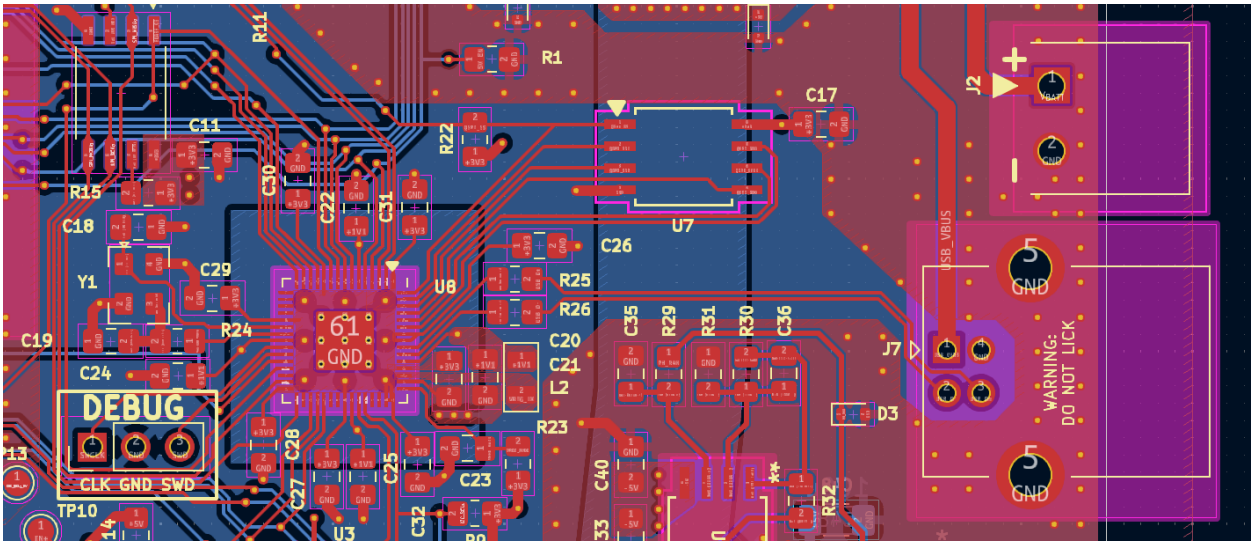


Figure 2: RP2350 Microcontroller Layout, as Produced in KiCAD

Next, the signal conditioner subsystem was routed. Important considerations for this section included prioritizing the I^2C traces in the routing order for reliable communication with the digital potentiometers and ADC. Analog traces needed to be kept short, while also maintaining access to test points for hardware debugging.

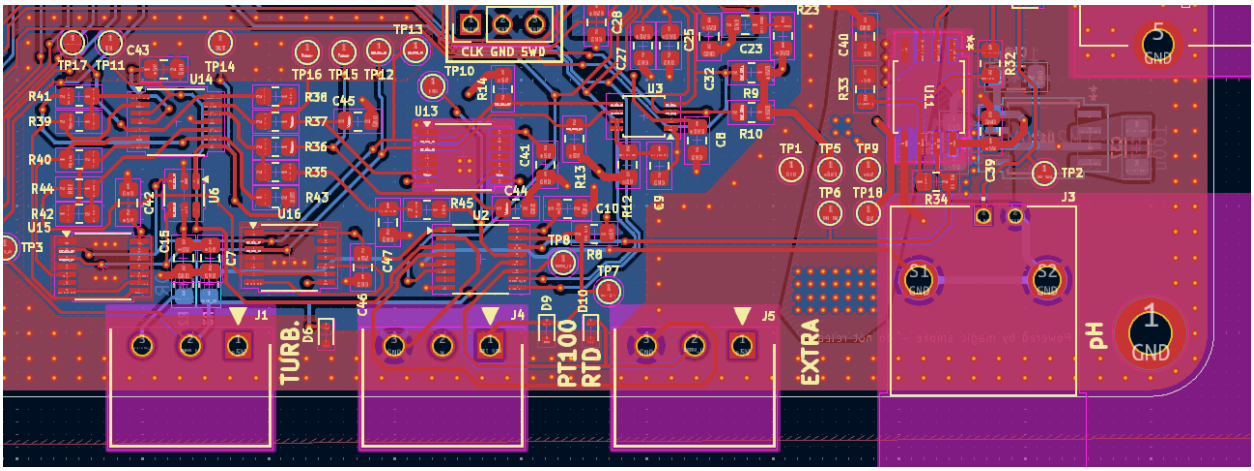


Figure 3: Signal Conditioner Subsystem Layout, as Produced in KiCAD

The final routing section is the RF interface, for which most of the components were managed by the integration of the Waveshare Core1262 LoRa Modulator Module. Key considerations in this section included maintaining 50Ω characteristic impedance along the coplanar waveguide, integrating pi filter footprints such that they have minimal impact on trace geometry, and utilizing fencing and stitching vias in this area as well as along the board edge to minimize noise and interference caused by this section.

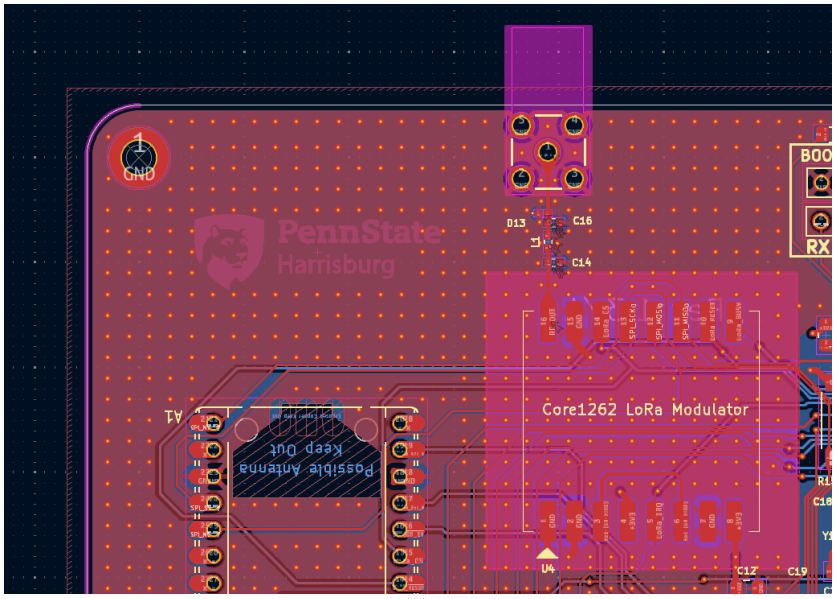


Figure 4: RF Frontend Layout, as Produced in KiCAD

PCB Photos

Below are 2D plots and 3D renders of the final PCB design, as produced in KiCAD. In addition to the layout described in the previous section, a full Raspberry Pi Pico 2 Footprint was added to the design as a failsafe mechanism if any layout errors in the microcontroller system of the board would prevent the functioning of the design.

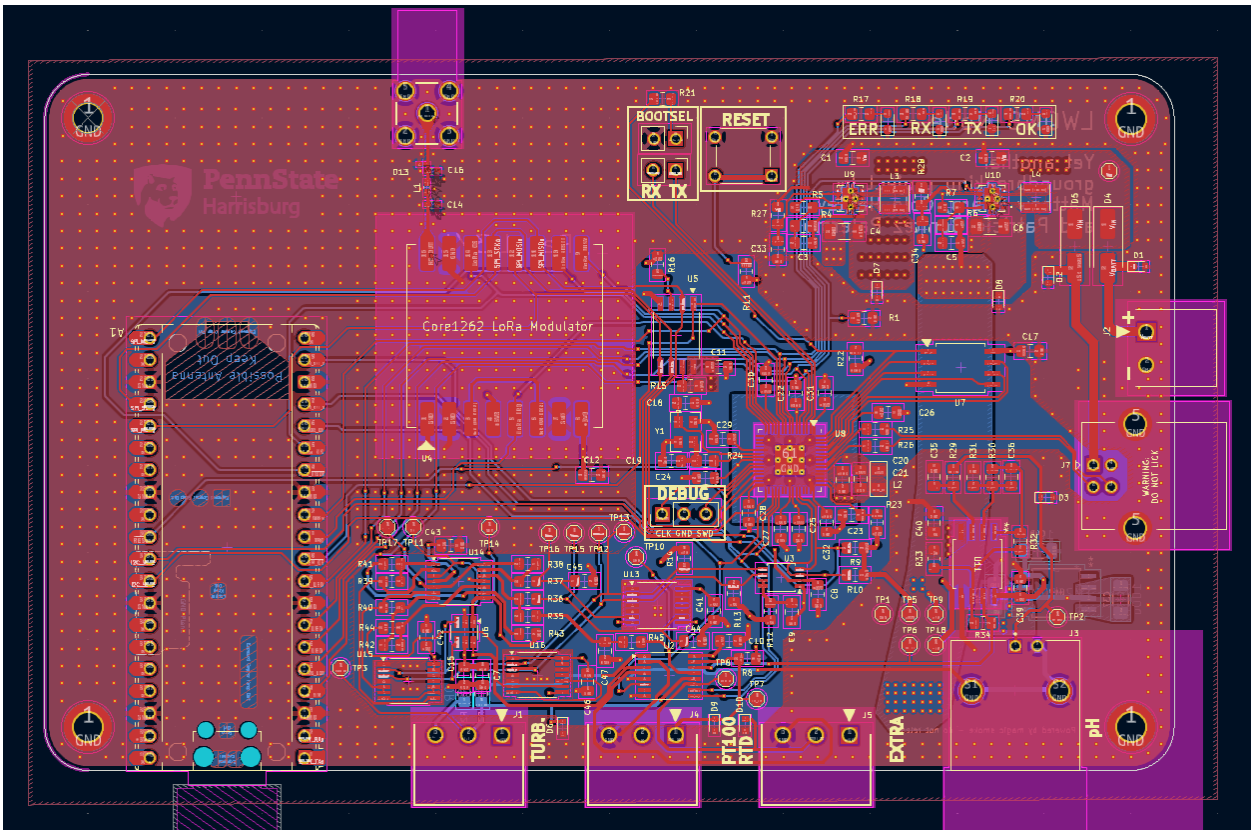


Figure 5: Full 2D Render of Design as produced in KiCAD. Internal layers are hidden to show tracks on signal layers.

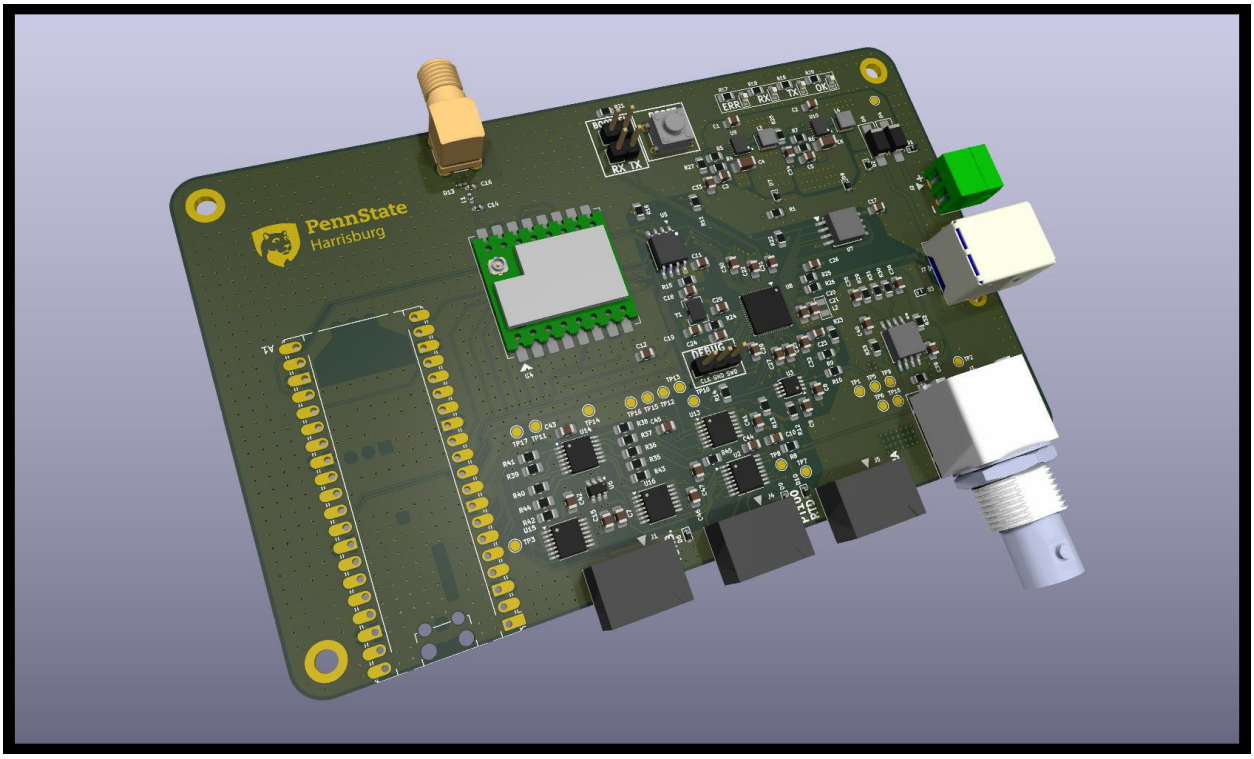


Figure 6: Full 3D Render of PCB, Populated

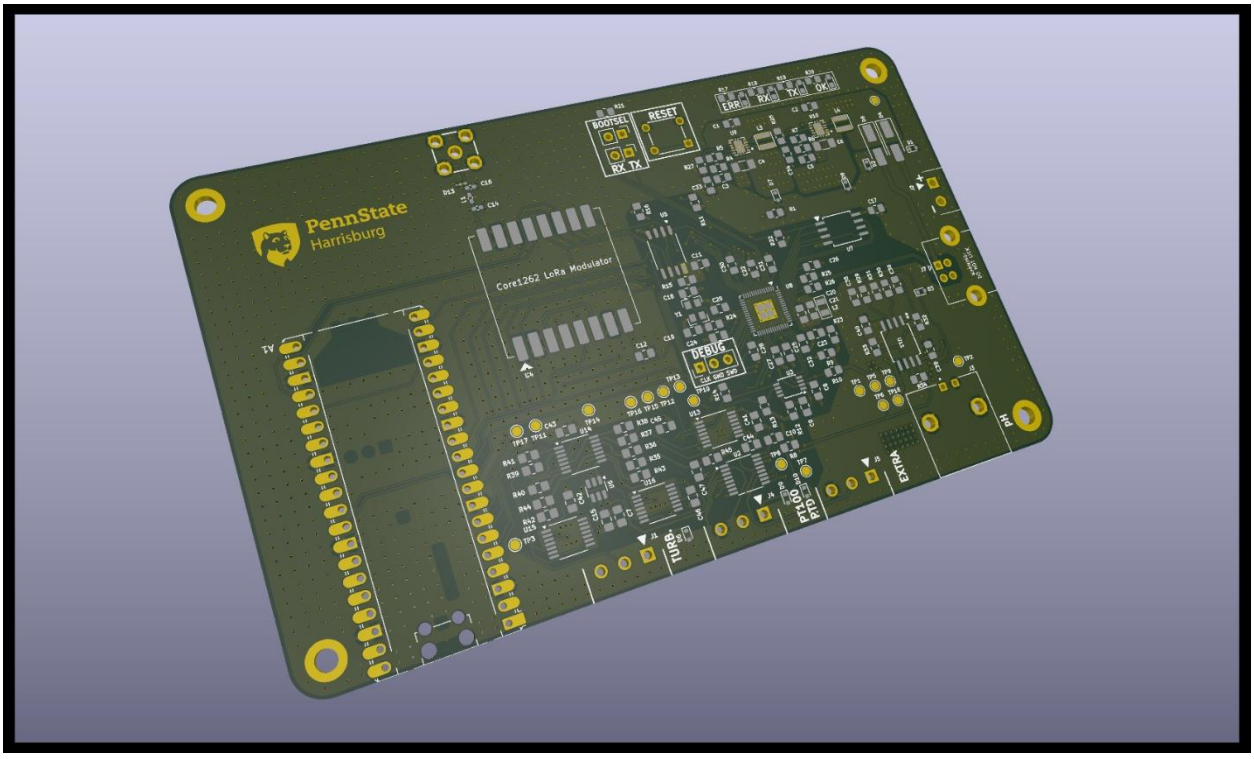
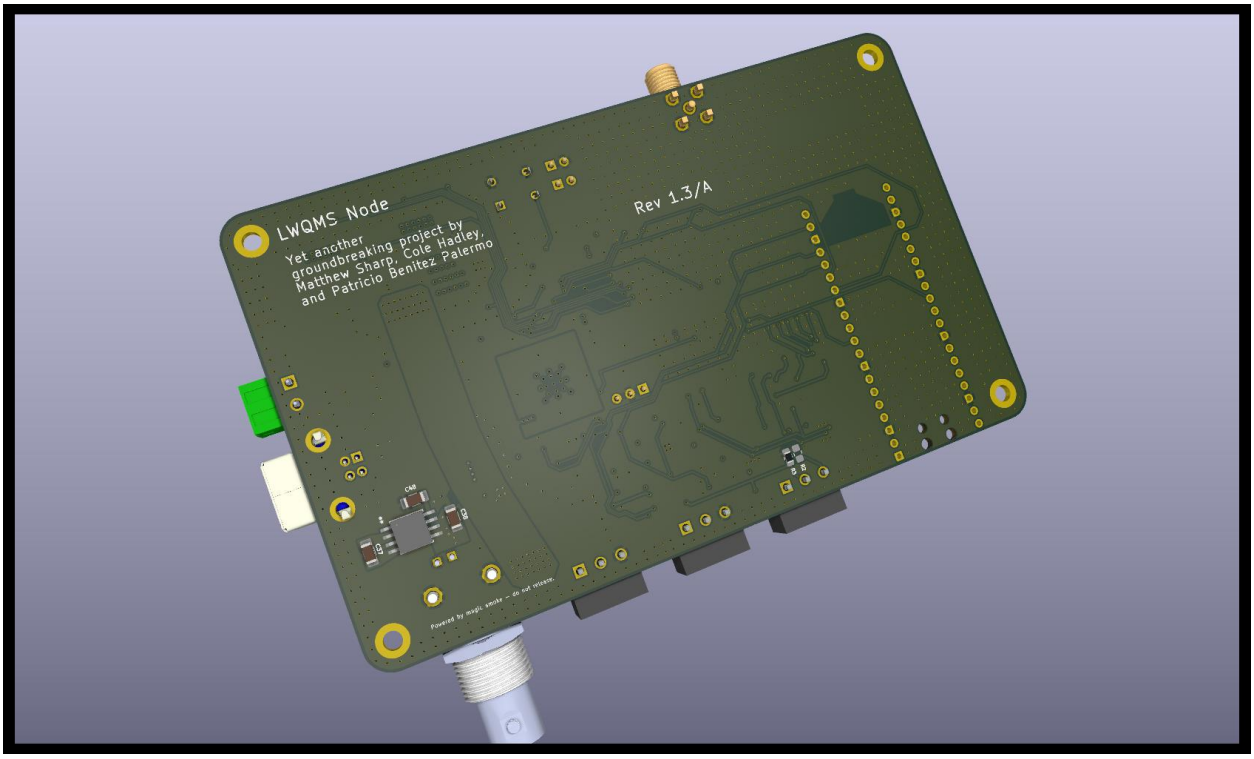
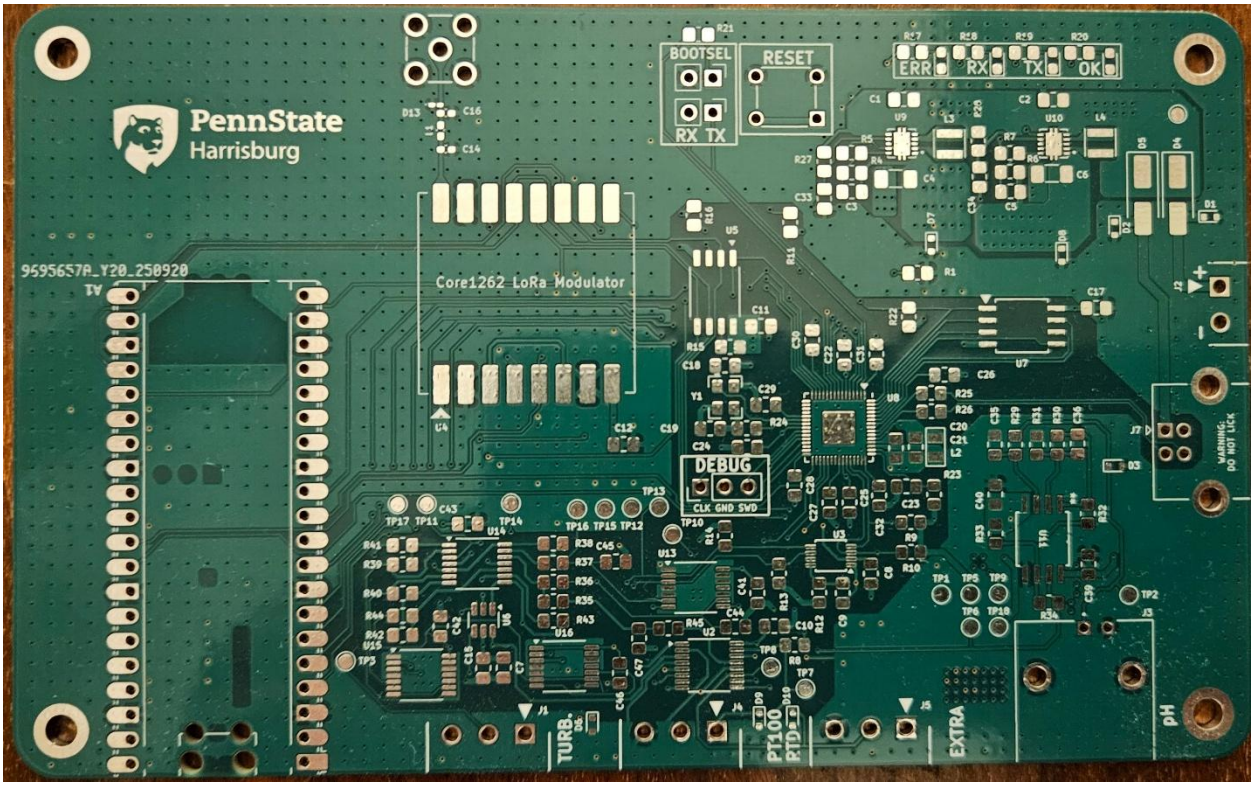


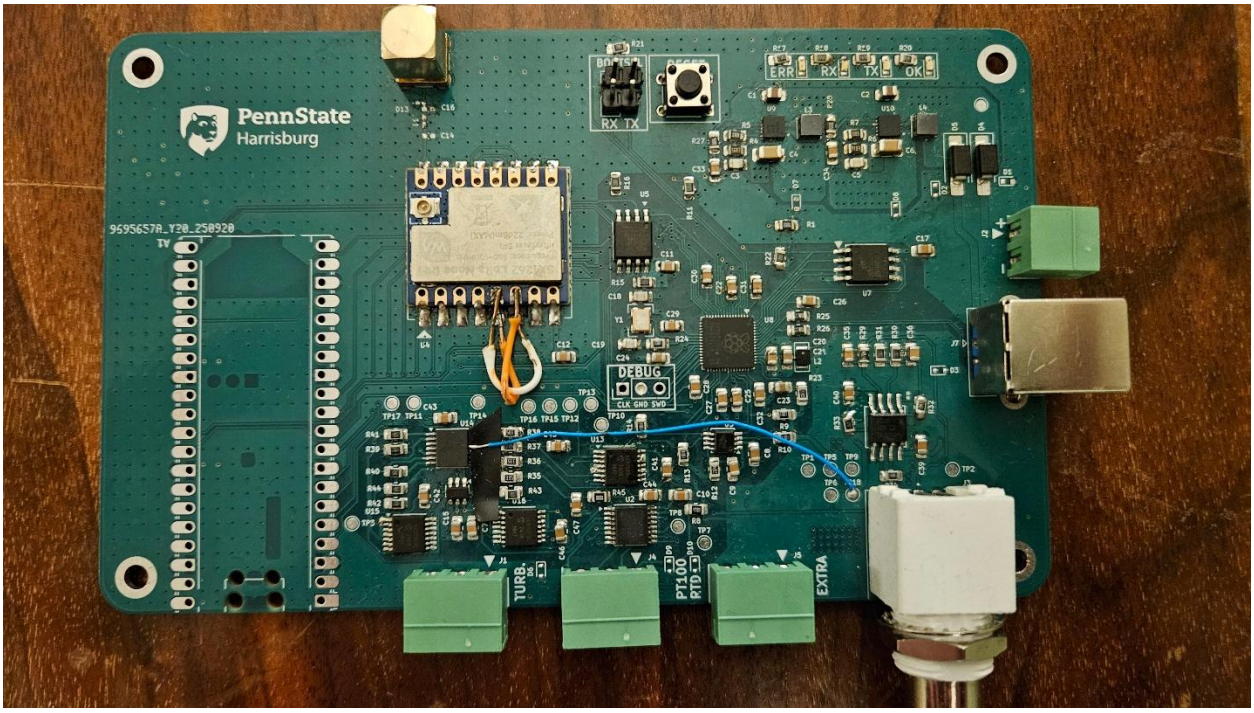
Figure 7: Full 3D Render of PCB, Unpopulated



3D Render of Reverse Side of PCB, Populated



Final Ordered PCB, Unpopulated



Final Ordered PCB, Populated

PCB Testing & Discussion of Results

Testing Background

To account for nonlinearities and resistance deviations between digital potentiometers discovered during initial testing, an automatic calibration routine was incorporated into the sensor node's firmware. The calibration measures the DC offsets and Gain Resistor values at each potentiometer wiper step location (0-256). This data is then stored on a NOR flash IC on the PCB, which the microcontroller then reads into memory, using it to accurately set the digital potentiometers according to desired gain and DC offset values. Plots of the DC offset and gain resistance values, as well as corresponding error are shown in *figures 8-11* below.

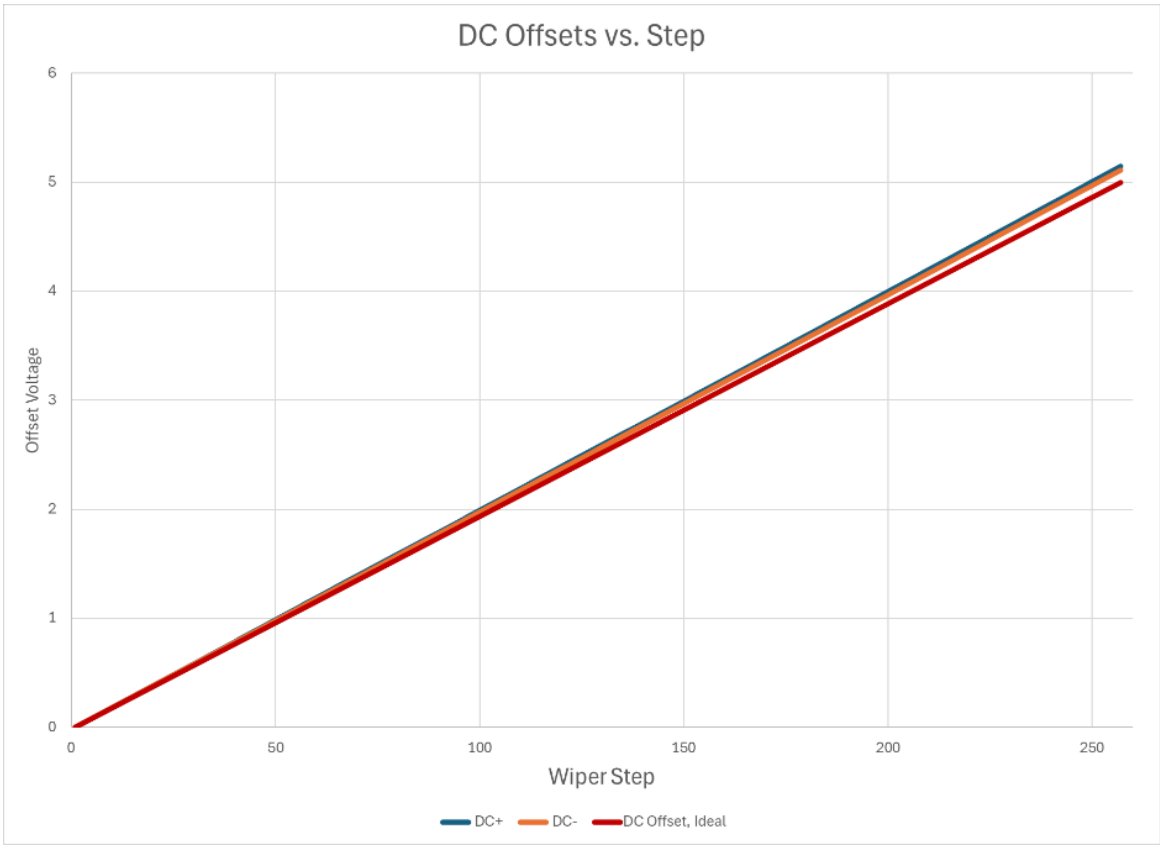


Figure 8: DC Offset Voltages versus Wiper Step Location. DC+ is plotted in Blue, DC- is plotted in Orange, and the Ideal DC Offset Voltage is Plotted in Red.

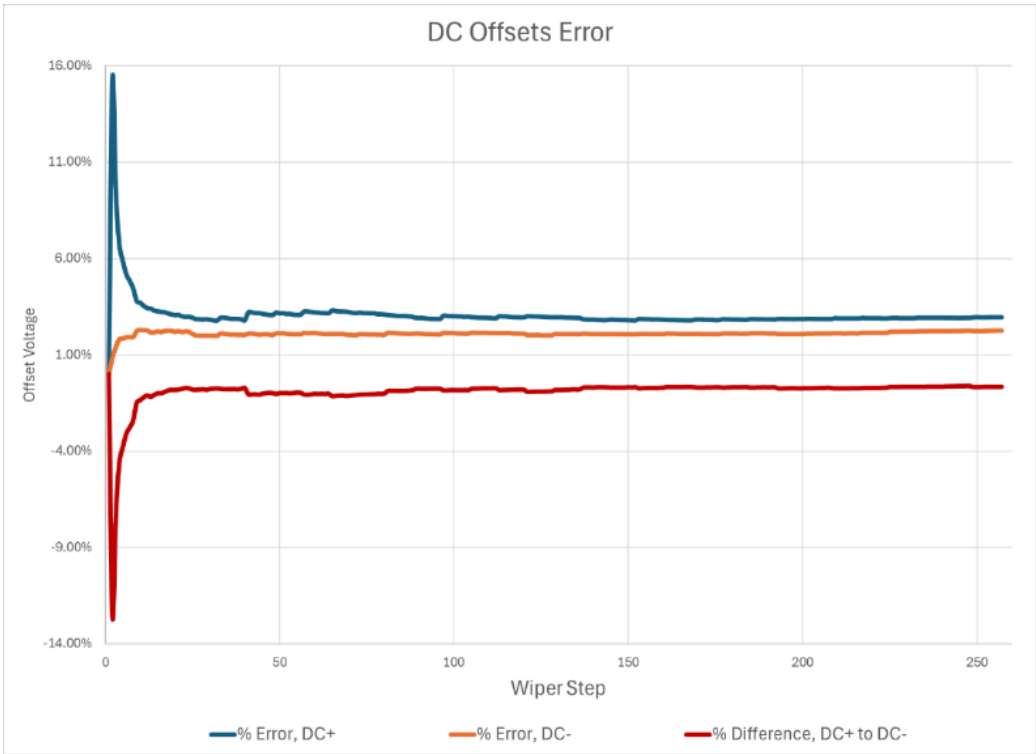


Figure 9: DC Offset Voltage Error with Respect to Ideal, and Each Other. The DC+ and DC- Error with Respect to the Ideal DC Offset Voltage are Plotted in Blue and Orange, Respectively, while the Difference between the Offsets is Plotted in Red.

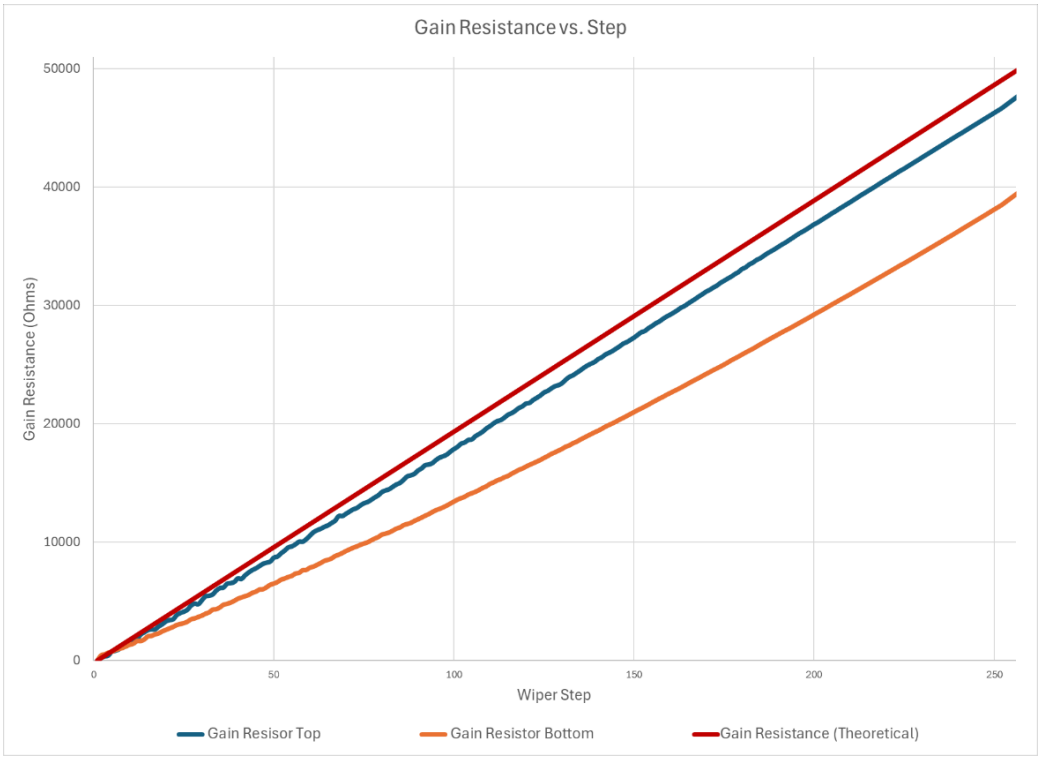


Figure 10: Gain Resistance Measurements Versus Wiper Step Location. The Top and Bottom Halves of the Gain Resistance are Plotted in Blue and Orange, Respectively, while the Theoretical Gain Resistance is Plotted in Red.

Figure 11: Gain Resistance Error. The error with respect to the theoretical gain resistance for the top and bottom halves of the gain resistor are plotted in red and orange, respectively, while the difference between the two resistors is plotted in green.

Testing Methods & Relation to Engineering Requirements

With the calibration of the software-defined instrumentation amplifier portion of the signal conditioner subsystem complete, a preliminary test of the calibration software component was performed. Following from the engineering requirements of the signal conditioner subsystem:

- Interprets the analog voltages produced by the sensors as digital data
- Converts the digitized analog voltage readings to relevant measurement units for pH, turbidity, and temperature
- Formats the data to be sent to the central gateway module

Accurate and reliable operation of the software-defined instrumentation amplifier, which pre-processes incoming analog sensor voltages to be compatible with the ADC and provide increased resolution in the measurement region of interest, are necessary for full operation of this subsystem.

Tables 1 and 2 below show the performance of the software-defined instrumentation amplifier in meeting desired analog characteristics for both gain and input

DC offset. The values selected by the firmware using the calibration data were then compared against the resulting values if an idealized potentiometer was considered in the code to compute both the gain and the DC offsets. For example, if a gain of 3 is requested by the user or within an autonomous call to the function in the code, the firmware will select the combination of gain resistors on both the top and bottom wipers that are as close to $10k\Omega$ as possible $\left(Gain = \frac{2\times10k\Omega}{R_G} + 1 = 3 \rightarrow R_G = 10k\Omega\right)$. If this calibration data was not used, the firmware would instead need to select the wiper positions assuming an ideally linear potentiometer with end-to-end resistance of $50k\Omega$. For a gain of 3, the selected wiper position would be calculated as: $\left(1 - \frac{10k\Omega}{50k\Omega}\right) * 256 = 205$, producing a gain resistance of $9.8132k\Omega$ given the measured resistance of the potentiometer at that location, and a resulting gain of 3.03, introducing non-negligible error into the system. The resulting performance difference in the amplifier output voltage and computed input voltage by the firmware for both the calibration-controlled system and the non-calibrated system are shown in *tables 3, and 4* below.

As seen in the tables below, the integration of calibration data into the software-defined instrumentation amplifier greatly improves system performance, driving gain to 0.26% in the worst case compared to 11.63% in the worst case for the uncalibrated system, and ultimately resulting in significant reductions in error between the calculated input voltage of the system and the actual input voltage, demonstrating the software component of the design's ability to help the system meet its functional requirements.

Table 1: Gain Selection Performance of the Software-Defined Calibration Data, Uncalibrated vs. Calibrated

Input Voltage	Ideal Gain	Actual Gain, Assuming Ideal Pots	% Error	Software-Selected Gain	% Error
0.10727	12	10.64	11.33%	12.030886	0.26%
0.40825	4	3.95	1.25%	4.000105	0.00%
0.7996	2	2.01	0.50%	2.000037	0.00%
1.20286	3.5	3.469	0.89%	3.49919	0.02%
1.5077	3	3.0022	0.07%	2.99962	0.01%
1.8005	5	4.936	1.28%	4.998462	0.03%
2.00948	2.5	2.513	0.52%	2.500217	0.01%

Table 2: DC Offset Selection Performance of the Software-Defined Instrumentation Amplifier, Uncalibrated vs. Calibrated

Input Voltage	Ideal DC Offset	Actual DC Input Offset, Assuming Ideal Pots	% Error	Software-Selected DC Offset	% Error
0.10727	0	0.00002512	N/A	0.002512	N/A
0.40825	0	0.00002512	N/A	0.002512	N/A
0.7996	0	0.00002512	N/A	0.002512	N/A
1.20286	-1	-0.977	2.30%	-0.994864	0.51%
1.5077	-1	-0.977	2.30%	-0.994864	0.51%
1.8005	-1.5	-1.534	2.27%	-1.492605	0.49%
2.00948	-1.5	-1.534	2.27%	-1.492605	0.49%

Table 3: Output Voltage Performance of the Software-Defined Instrumentation Amplifier, Uncalibrated vs. Calibrated

Input Voltage	Theoretical ADC Voltage	Measured ADC Voltage	% Error, w.r.t. Theoretical	ADC Voltage, Assuming Ideal Pot	% Error, w.r.t. Theoretical	% Difference, Measured vs. Idealized
0.10727	1.2	1.425812	18.82%	1.141620077	4.86%	19.93%
0.40825	1.6	1.703875	6.49%	1.612686724	0.79%	5.35%
0.7996	1.6	1.602	0.13%	1.607246491	0.45%	0.33%
1.20286	0.7	0.7775	11.07%	0.78350834	11.93%	0.77%
1.5077	1.5	1.598125	6.54%	1.59326754	6.22%	0.30%
1.8005	1.5	1.639438	9.30%	1.315444	12.30%	19.76%
2.00948	1.25	1.321938	5.76%	1.19488124	4.41%	9.61%

Table 4: Input Voltage Calculation for the Software-Defined Instrumentation Amplifier, Calibrated vs. Uncalibrated

Input Voltage	Actual ADC Voltage	Calculated Input Voltage, Assuming Ideal Pots	% Error, w.r.t. Actual	Software-Estimated Actual Input Voltage	% Error, w.r.t. Actual
0.10727	1.425812	0.133979767	24.90%	0.11463	6.86%
0.40825	1.703875	0.431335639	5.65%	0.419321	2.71%
0.7996	1.602	0.796989805	0.33%	0.790224	1.17%
1.20286	0.7775	1.201127991	0.14%	1.212298	0.78%
1.5077	1.598125	1.509317967	0.11%	1.522139	0.96%
1.8005	1.639438	1.866138979	3.65%	1.817292	0.93%
2.00948	1.321938	2.060039793	2.52%	2.014735	0.26%

With the calibration data obtained for the amplifier and the signal acquisition process validated, the final testing step was using the amplifier to convert incoming analog sensor voltages to meaningful measurements for water quality management. Shown below in *figures 12, 13, and 14* are plots corresponding to the input voltage versus environmental measurement value for the pH sensor, RTD (temperature) sensor, and turbidity sensor, respectively. During testing of the RTD, a consistent $+3mV$ offset was observed on the input voltage compared to the expected voltage measurement. To account for this, a $-3mV$ offset was applied to the RTD input; since $3mV$ is too fine of an offset to apply using the software-defined instrumentation amplifier, this offset was applied in software. This change is reflected in *figure 13* below, showing the raw temperature measurement in blue, the actual environmental temperature in orange, and the calculated temperature in green after the $-3mV$ input offset was applied. As shown in the figure, applying this offset produced a temperature response much more consistent with the ideal response.

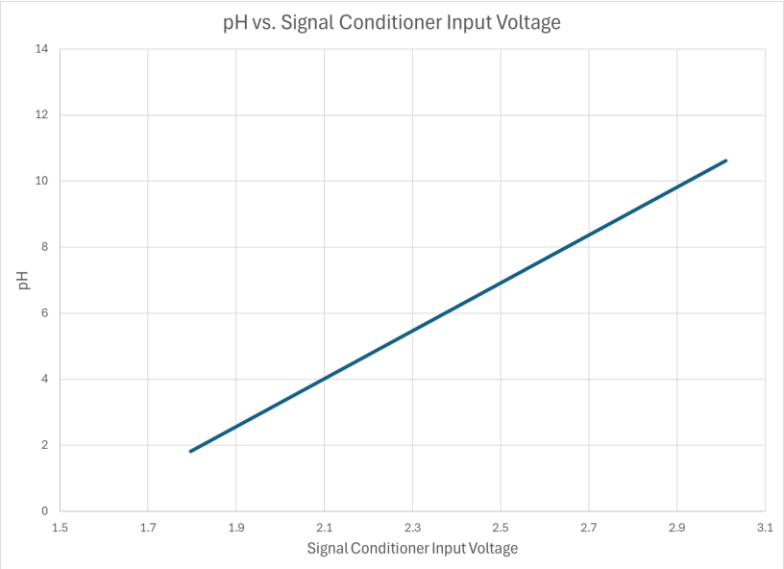


Figure 12: pH Sensor Voltage as measured by the Signal Conditioner versus Actual pH.

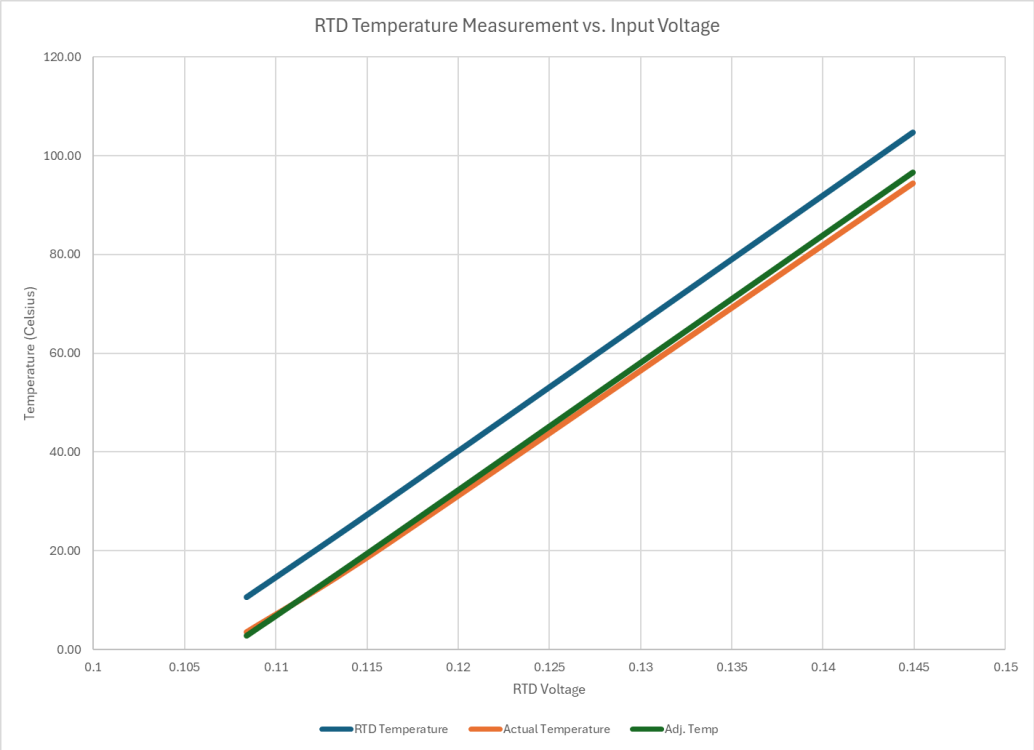


Figure 13: RTD Temperature vs. Input Voltage

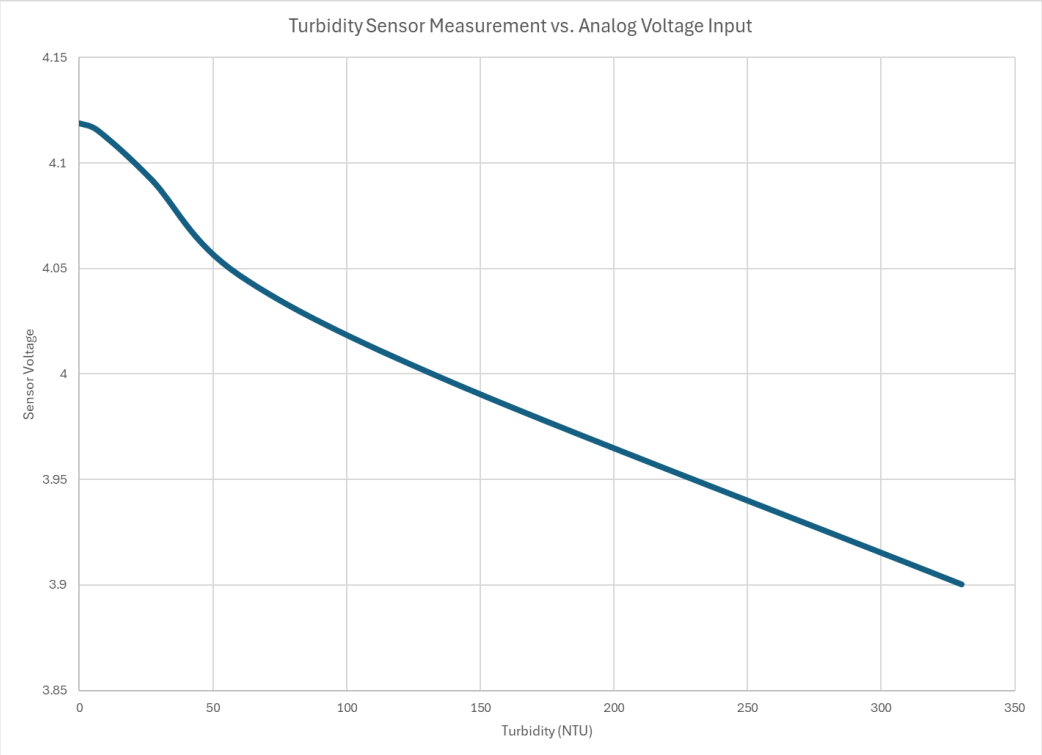


Figure 14: Turbidity Sensor Measurement vs. Input Voltage