UNIVERSITY OF COLORADO - BOULDER

ECEN 5730
PRACTICAL PCB DESIGN MANUFACTURE — FALL 2024

Board 4 Report - Instrument Droid

Sam WALKER

Tim Swettlen

Monday, December 16, 2024

College of Engineering & Applied Science
UNIVERSITY OF COLORADO BOULDER

Introduction

This report presents the design, testing, and analysis results for **Board 4**, referred to as the "Instrument Droid." This custom board serves as an upgraded, PCB-based implementation of the solderless breadboard VRM Characterizer Instrument Droid previously demonstrated in Labs 21 and 22. The focus of this project was to refine the VRM characterization tool for enhanced accuracy, durability, and feature integration, including DAC/ADC control, MOSFET switching, smart LEDs, and buzzer feedback. Key metrics analyzed include bootloading, signal integrity, MOSFET operation, current measurement accuracy, and power regulation under dynamic loads.

Plan of Record (POR)

The Plan of Record for **Board 4** involves verifying the functionality of the Instrument Droid and comparing its performance to the earlier solderless breadboard implementation. The following objectives guide the testing process:

- **Bootloader Verification**: Ensure successful bootloading of the ATmega328, confirming communication and programming capability.
- **DAC and ADC Operation**: Validate the DAC output and ADC measurement accuracy using controlled test signals.
- **MOSFET Functionality**: Assess MOSFET switching behavior under dynamic loads and verify safe operating conditions.
- **Thevenin Resistance Measurement**: Demonstrate the ability to measure voltage, current, and calculate Thevenin resistance for various voltage sources.
- **Buzzer and LED Functionality**: Confirm smart LEDs and buzzer indicate the progress and results of automated tests.
- **Noise and Power Analysis**: Evaluate signal integrity, switching noise, and power consumption under various test scenarios.

Risk Reduction

The PCB-based implementation introduces new challenges related to layout, component placement, and heat management. Special attention is given to:

- Ensuring signal and power integrity by carefully routing high-speed and sensitive traces.
- Minimizing thermal stress on the MOSFET and other power components through proper design and testing.
- Debugging unforeseen component issues (e.g., DAC behavior) to maintain reliable operation.

Project Overview

Block Diagram

The block diagram illustrates the flow of power and signals within the Instrument Droid, including the DAC-controlled MOSFET driver, ADC feedback loop, and smart LED/buzzer indicators.

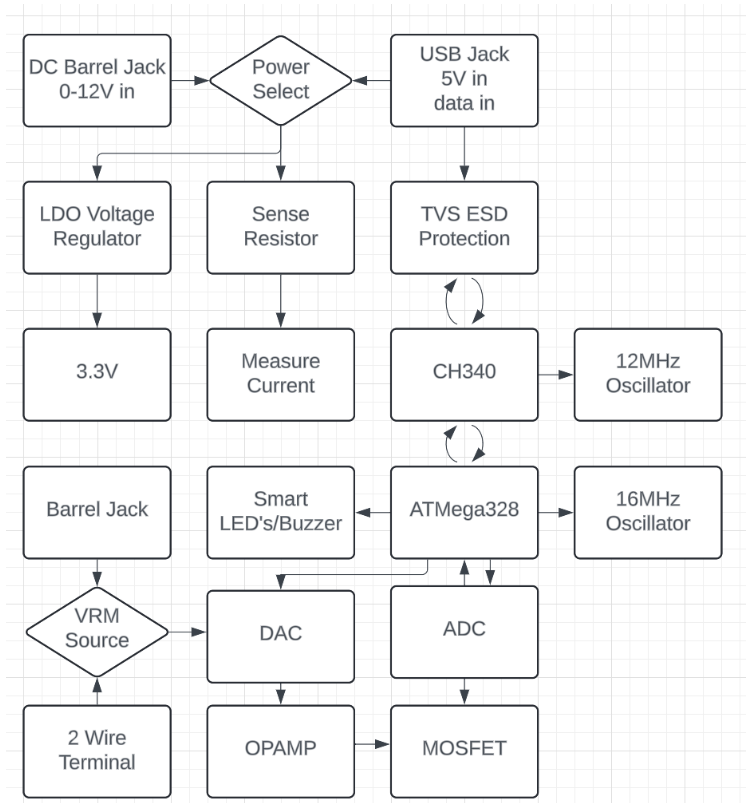

Figure 1: Block Diagram of Instrument Droid

Schematic in Altium Designer

The schematic demonstrates the interconnection of key components, including the ATmega328, MCP4725 DAC, ADS1115 ADC, MOSFET, and supporting circuits for power regulation and signal routing.

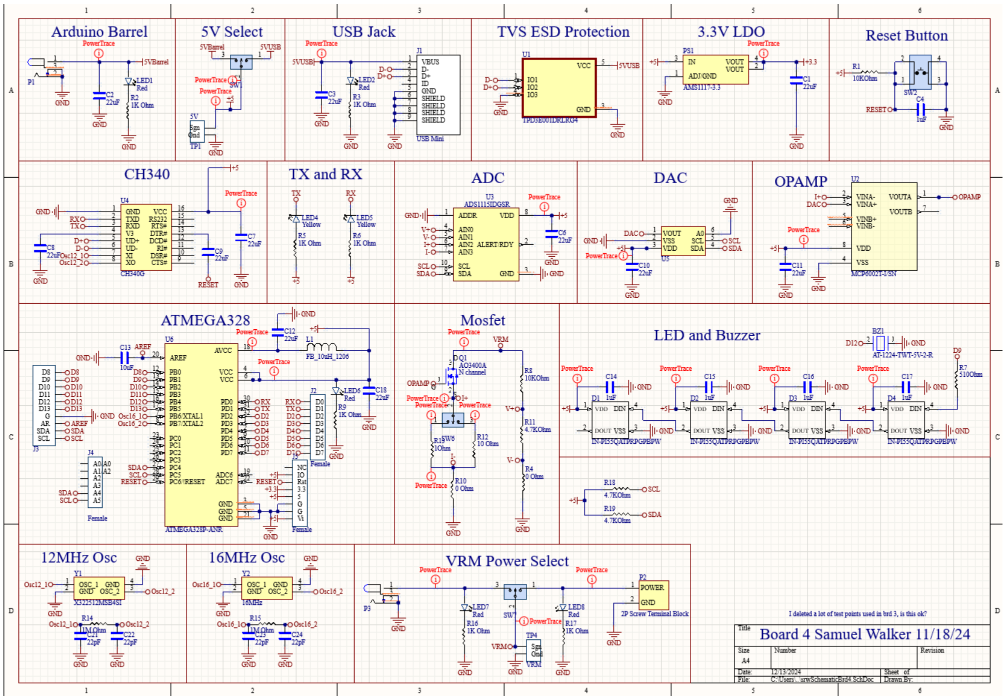

Figure 2: Schematic in Altium Designer

PCB Layout

The PCB features a 4-layer design with components on both sides to optimize space and performance. The layout prioritizes clean signal routing, thermal management, and EMI reduction.

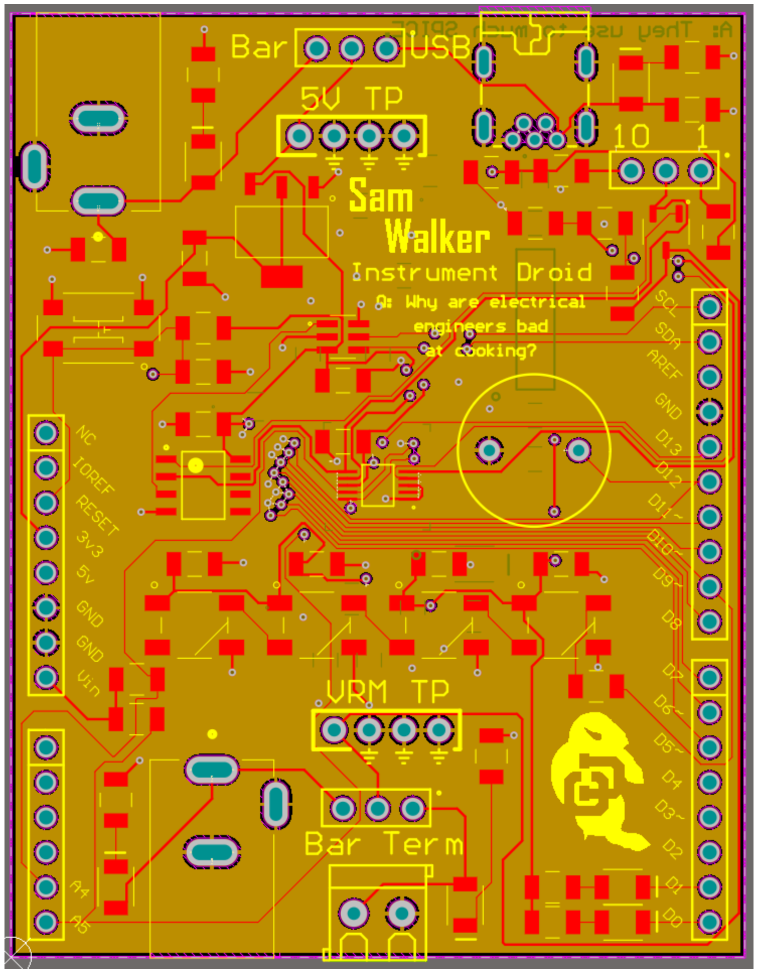

Figure 3: PCB Layout - Front Layer

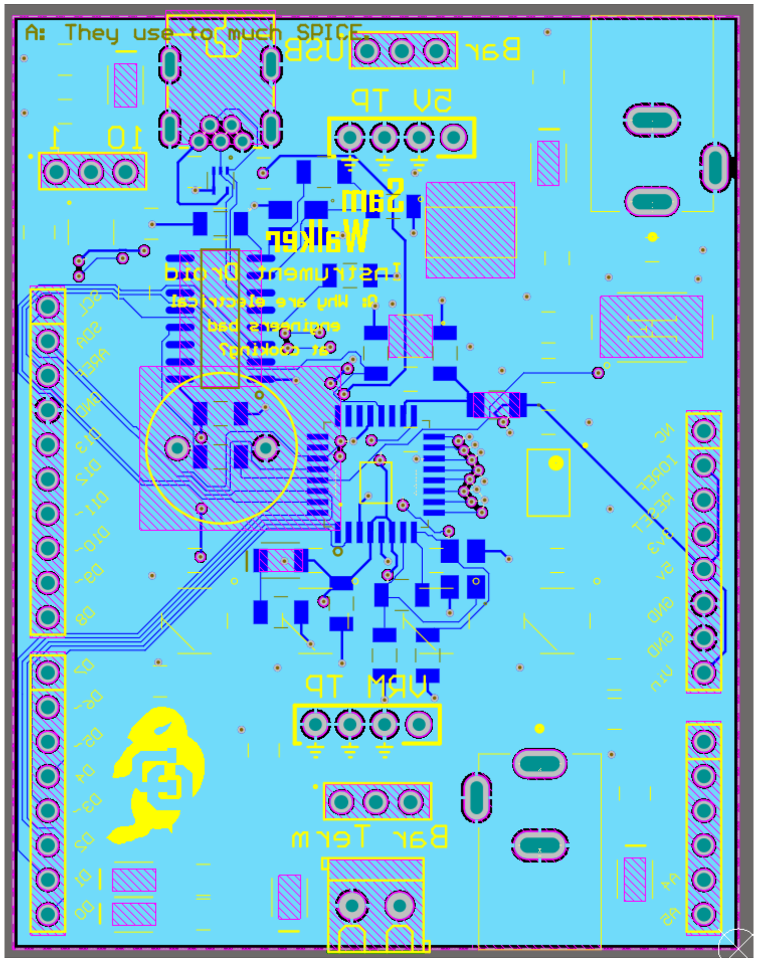

Figure 4: PCB Layout - Back Layer

Unassembled and Assembled PCB

The unassembled PCB, manufactured by JLCPCB, highlights the precision of the design. The assembled PCB includes all components, carefully soldered and verified for proper alignment and connectivity.

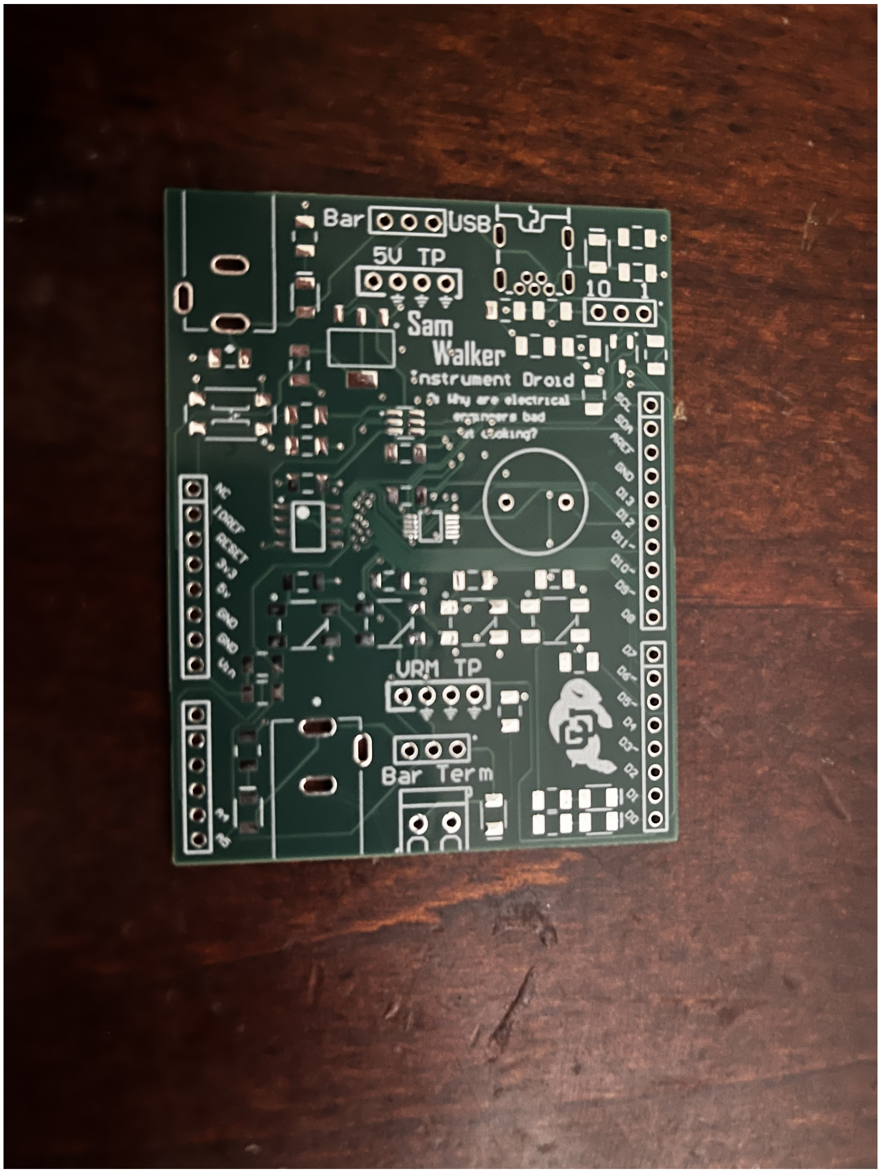

Figure 5: Unassembled PCB from JLCPCB

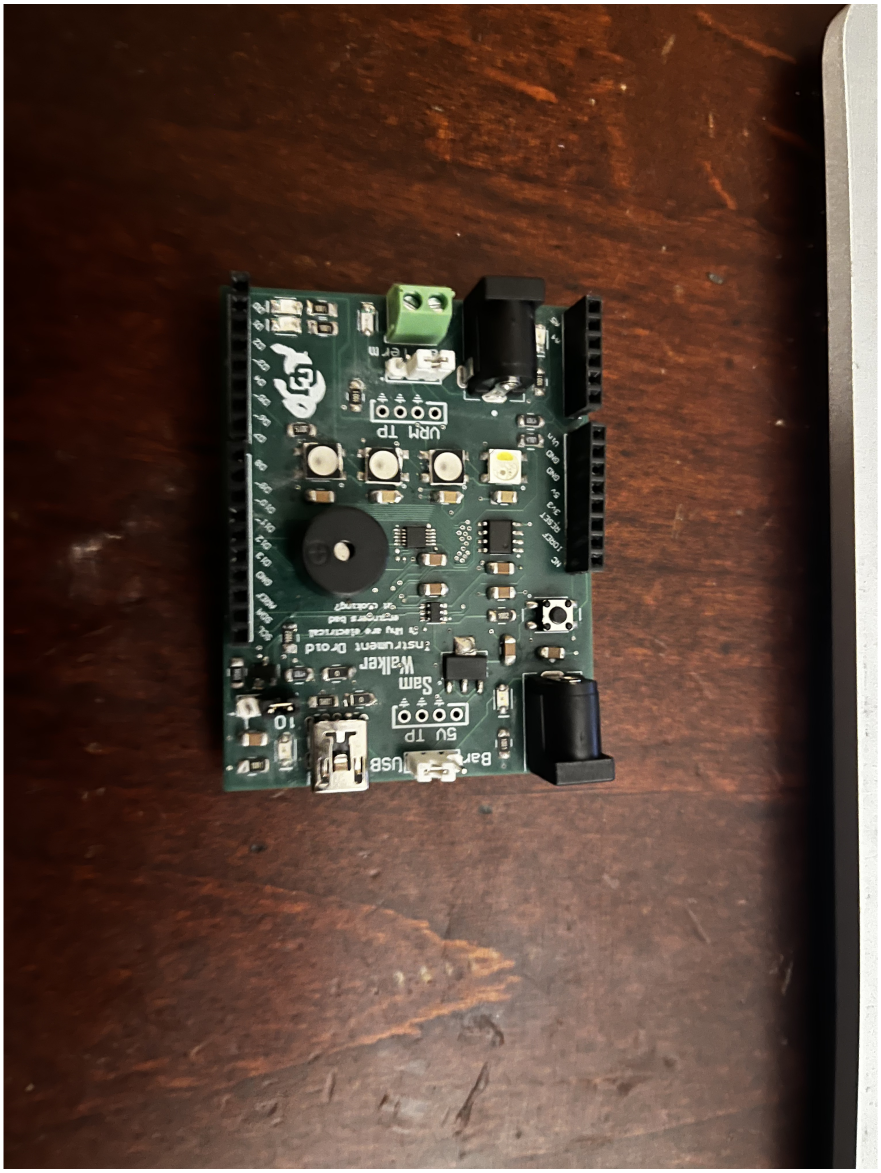

Figure 6: Assembled Instrument Droid PCB

Testing and Measurements

Bootloading

Bootloading the ATmega328 on Board 4 was successfully demonstrated. The process required manually connecting individual pins due to the absence of ICSP headers. This added complexity, particularly with the reset pin connection, but the issue was resolved after uploading the ArduinoISP script to the programmer board. The board responded to commands as expected post-bootloading.

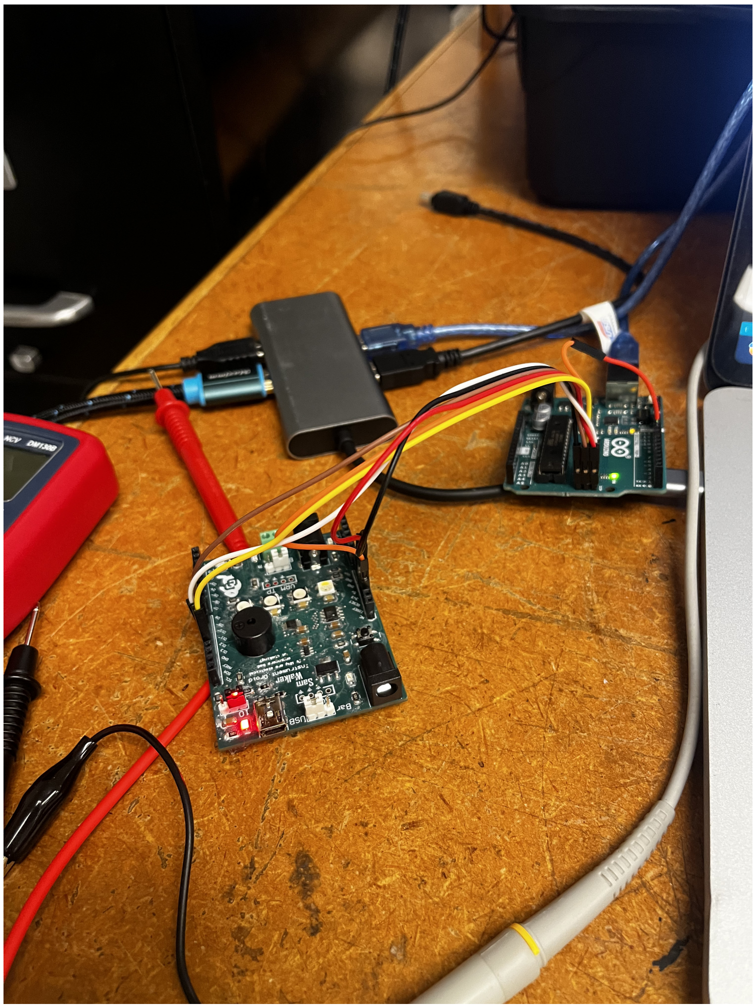

Figure 7: Bootloading Setup with Arduino UNO

Uploading Code - Example Blink Sketch

To verify programming capability, a simple blink sketch was uploaded to the ATmega328. The sketch toggled pin 13 with a 100 ms delay, creating a 200 ms period. This behavior was confirmed using an oscilloscope, as shown below.

```
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(100);
  digitalWrite(LED_BUILTIN, LOW);
  delay(100);
}
```

Figure 8: Blink Sketch Code Example and Resulting Signal

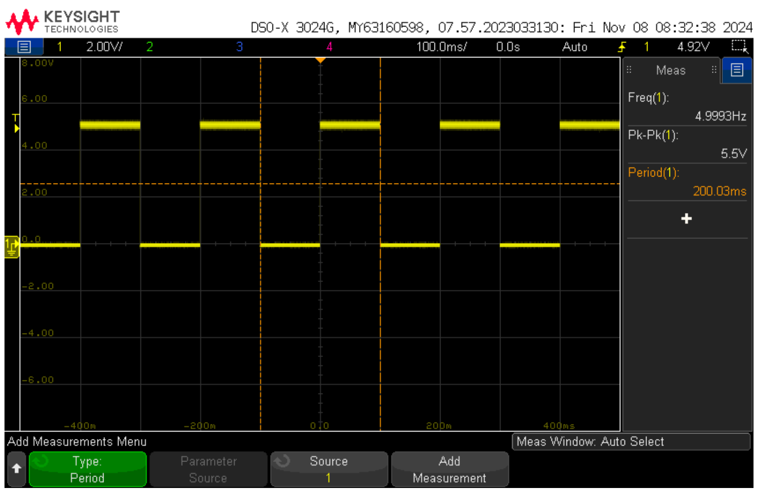

Figure 9: Digital Pin 13 Blinking Signal

DAC Output

The DAC output was validated using a test script designed to produce a controlled signal. The oscilloscope captured the output waveform, confirming proper operation of the DAC.

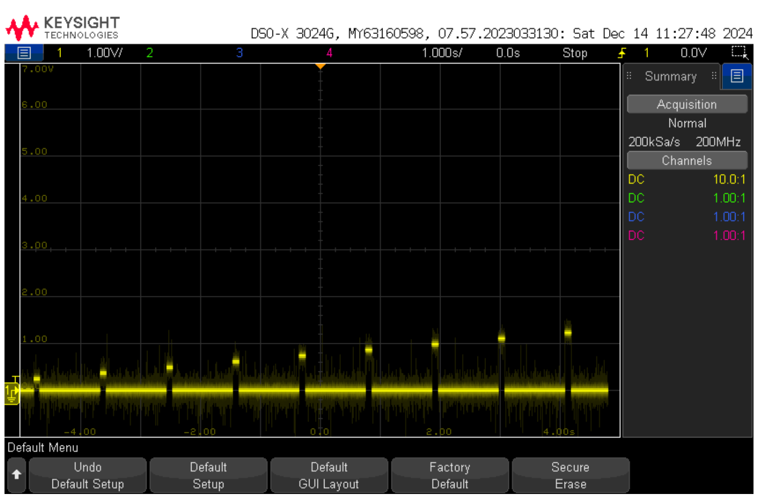

Figure 10: DAC Output Signal

MOSFET Operation

The MOSFET's performance was tested under varying DAC outputs. The first figure below shows the MOSFET's switching behavior. The sense resistor voltage, captured in the following figure, measured approximately 2V. Given a 10Ω sense resistor, the current was calculated as:

$$I = \frac{V}{R} = \frac{2V}{10\Omega} = 200\,\text{mA}$$

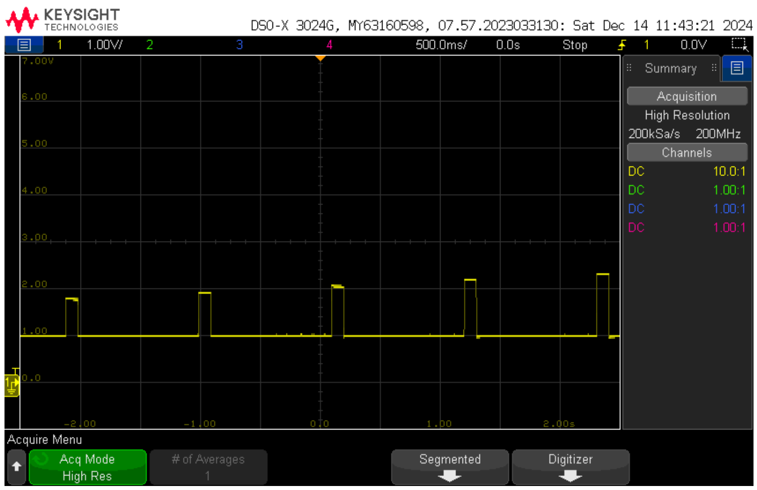

Figure 11: MOSFET OpAmp Signal

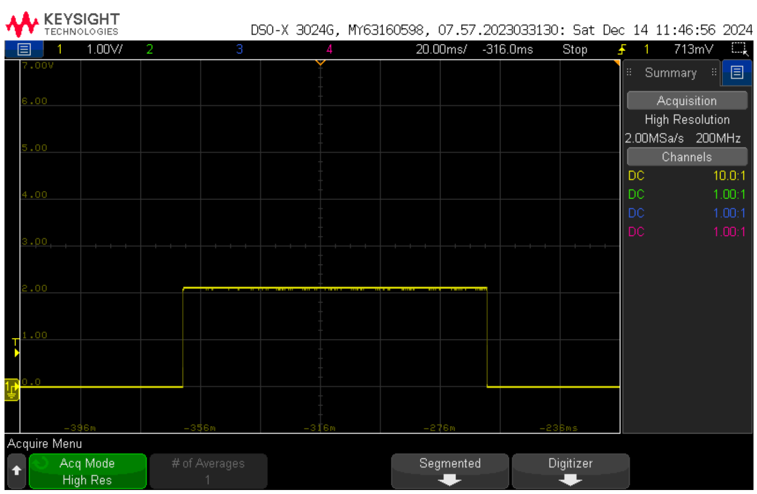

Figure 12: Sense Resistor Signal (2V)

Thevenin Resistance Measurement

The board was used to measure the Thevenin resistance of voltage sources. For wall warts tested, the calculated Thevenin resistance was approximately 0.4Ω. Voltage and current measurements for 5V and 9V sources, shown in the figures below, supported this conclusion.

```
20:25:16.185 -> 1, 12.288, 5.2092, 5.2049, 0.3480
20:25:17.261 -> 2, 24.536, 5.2088, 5.1984, 0.4237
20:25:18.382 -> 3, 36.894, 5.2088, 5.1955, 0.3594
20:25:19.505 -> 4, 49.325, 5.2086, 5.1890, 0.3972
20:25:20.626 -> 5, 61.647, 5.2083, 5.1868, 0.3487
20:25:21.703 -> 6, 74.420, 5.2084, 5.1814, 0.3625
20:25:22.828 -> 7, 86.657, 5.2088, 5.1753, 0.3873
20:25:23.953 -> 8, 98.998, 5.2089, 5.1707, 0.3852
20:25:25.030 -> 9, 111.124, 5.2090, 5.1686, 0.3637
20:25:26.157 -> 10, 123.223, 5.2087, 5.1603, 0.3932
20:25:27.237 -> 11, 135.908, 5.2089, 5.1565, 0.3858
20:25:28.366 -> 12, 148.185, 5.2084, 5.1513, 0.3859
20:25:29.490 -> 13, 160.469, 5.2088, 5.1492, 0.3714
20:25:30.568 -> 14, 172.769, 5.2093, 5.1424, 0.3875
20:25:31.691 -> 15, 185.080, 5.2086, 5.1383, 0.3797
20:25:32.814 -> 16, 197.462, 5.2081, 5.1348, 0.3711
20:25:33.892 -> 17, 209.600, 5.2091, 5.1272, 0.3910
20:25:35.013 -> 18, 221.759, 5.2082, 5.1230, 0.3842
20:25:36.138 -> 19, 234.064, 5.2090, 5.1196, 0.3822
20:25:37.214 -> 20, 246.368, 5.2086, 5.1143, 0.3826
20:25:37.214 -> done
```

Figure 13: 5V Test Results

```
20:24:11.297 -> 1, 12.322, 9.2253, 9.2488, -1.9054
20:24:12.419 -> 2, 24.540, 9.2255, 9.2463, -0.8470
20:24:13.540 -> 3, 37.009, 9.2250, 9.2409, -0.4291
20:24:14.615 -> 4, 49.294, 9.2255, 9.2381, -0.2561
20:24:15.737 -> 5, 61.738, 9.2253, 9.2307, -0.0870
20:24:16.859 -> 6, 74.274, 9.2258, 9.2217, 0.0552
20:24:17.934 -> 7, 86.666, 9.2268, 9.2116, 0.1761
20:24:19.058 -> 8, 98.858, 9.2260, 9.2079, 0.1832
20:24:20.178 -> 9, 110.931, 9.2267, 9.2014, 0.2280
20:24:21.255 -> 10, 123.373, 9.2264, 9.1933, 0.2689
20:24:22.373 -> 11, 135.952, 9.2265, 9.1882, 0.2819
20:24:23.489 -> 12, 148.459, 9.2279, 9.1851, 0.2887
20:24:24.616 -> 13, 160.506, 9.2273, 9.1748, 0.3276
20:24:25.697 -> 14, 172.953, 9.2266, 9.1690, 0.3327
20:24:26.817 -> 15, 184.985, 9.2268, 9.1621, 0.3495
20:24:27.940 -> 16, 197.763, 9.2258, 9.1552, 0.3567
20:24:29.016 -> 17, 209.984, 9.2268, 9.1469, 0.3806
20:24:30.135 -> 18, 221.966, 9.2271, 9.1449, 0.3703
20:24:31.262 -> 19, 234.232, 9.2268, 9.1376, 0.3808
20:24:32.342 -> 20, 246.783, 9.2275, 9.1329, 0.3831
20:24:32.342 -> done
```

Figure 14: 9V Test Results

Smart LEDs and Buzzer

The smart LEDs and buzzer were demonstrated successfully, functioning as intended during testing. The smart LEDs provide clear visual feedback, with all four LEDs lighting green to indicate a successful test. In the event of a test failure, the LEDs flash red, ensuring immediate visibility of the issue.

The buzzer complements the visual feedback by emitting a distinct tone upon test completion, regardless of whether the result is a success or a failure. This auditory signal adds an extra layer of feedback, making it easier to identify test completion in noisy environments or when visual monitoring is not feasible.

The figure below illustrates the smart LEDs lit green during a successful test.

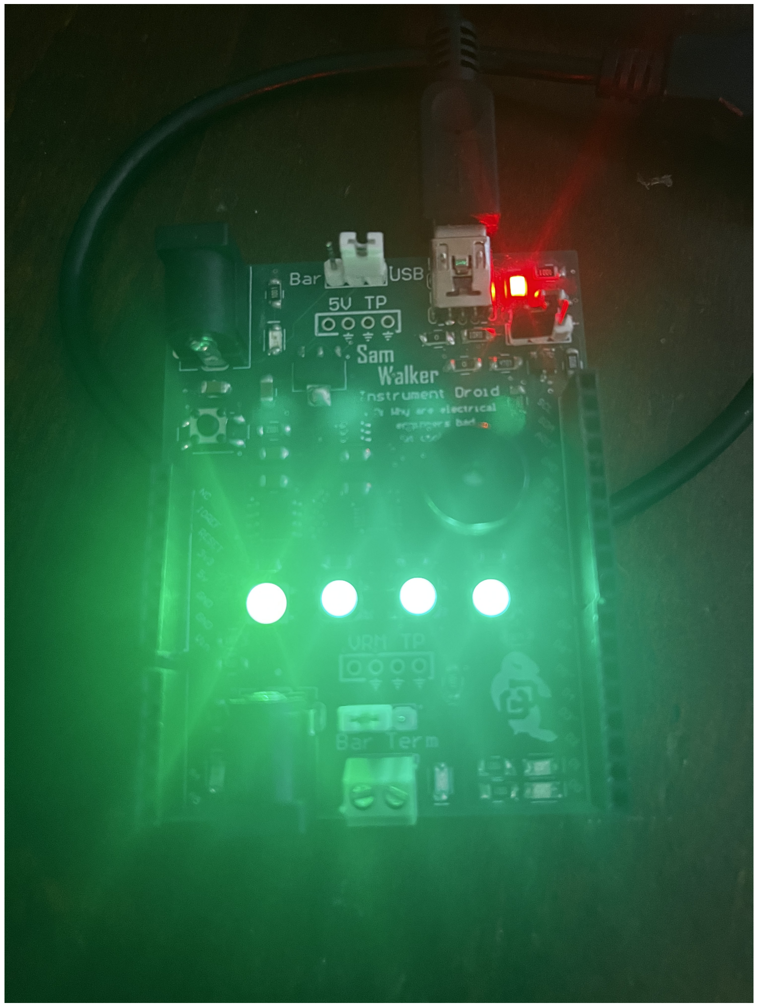

Figure 15: Smart LEDs During Testing

Summary of Learnings and Challenges

What Worked Well

- Successful bootloading and code uploading after troubleshooting and uploading the ArduinoISP script to the Arduino board.
- Reliable DAC and ADC operation achieved after replacing the faulty MCP4725 DAC.
- Smart LED and buzzer functionality provided clear and consistent feedback during testing.
- Overall thermal and EMI management performed well, with no significant electromagnetic interference detected during operation.

Challenges and Lessons Learned

- **Bootloading Oversight**: Initially failed to upload the ArduinoISP script to the Arduino board, which delayed bootloading. This was a simple oversight, but it reinforced the importance of a thorough pre-test checklist.
- **Faulty DAC Identification**: The initial MCP4725 DAC outputted a constant voltage due to a defect. After using a device identifier script, I replaced the DAC, resolving the issue. This experience emphasized the need for comprehensive component testing before integration.
- **Thermal Management**: The MOSFET reached temperatures exceeding 300C during testing, as the DAC was continuously outputting voltage. Script modifications resolved this, but this highlighted the importance of integrating thermal simulations early in the design process.
- **Component Compatibility**: Swapping the MCP4725 with an MCP4716 (a 10-bit DAC) initially caused confusion due to its identical footprint but different pinout. This experience underlined the importance of verifying not only component footprints but also their electrical connections and specifications.
- **I^2C Communication**: Unexpected behavior on the SDA and SCL lines initially raised concerns. However, further investigation revealed it was due to the reduced communication needs in the revised script. Monitoring I^2C activity with proper debugging tools helped confirm functionality.
- **Layout Considerations**: The absence of ICSP headers complicated bootloading. In future designs, dedicated headers will be included to streamline programming and debugging.

Improvements for Future Designs

- Include dedicated ICSP headers to simplify bootloading and programming.
- Perform more rigorous component testing, especially for critical ICs like DACs, to identify defects early in the prototyping phase.
- Implement a more thorough thermal management analysis, including simulations and heat dissipation strategies, to prevent overheating of components like MOSFETs.
- Verify not only the footprints but also the pinouts and electrical requirements of replacement components to avoid compatibility issues.
- Use scripting and debugging tools more extensively during testing to catch issues with communication lines and unexpected behavior early.

Conclusion

Board 4, the Instrument Droid, ultimately met its design objectives, showcasing reliable voltage regulator characterization and effective integration of key features like DAC, ADC, and feedback mechanisms. While challenges like a faulty DAC and MOSFET overheating were encountered, these issues were resolved through systematic troubleshooting and design modifications.

The lessons learned from this project—particularly in thermal management, bootloading, and component compatibility—will directly inform improvements in future designs. Overall, this project reaffirmed the value of custom PCB solutions for advanced instrumentation applications, setting a strong foundation for future iterations.