# Board 4 Report

## 4-Layer Instrument Droid Board based on Arduino

ECEN5730 Fall 2025

Hyounjun Chang

# Project Overview

## Plan of Record (POR)

- Do CDR of Golden Arduino (Board 4), which will be used as a base in Board 4 **(Completed Week 10)**
- Create Schematic for Instrument Droid (Board 4) **(Completed Week 11)**
- Test SBB version of Board 4 (ADC, DAC, Thevenin Resistance calculation) **(Completed Week 11)**
- Test and Bring-Up the Golden Arduino Board 3, which will also be used in Board 4 **(Completed Week 12)**
- Test Buzzer and WS2812B RGB LED strip **(Completed Week 13)**
- Do CDR of PCB Design of Instrument Droid (Board 4) **(Completed Week 13 with TA)**
- Bring-up PCB Board 4 post-assembly, and test functionalities as an Instrument droid **(Completed Week 16)**
- Write software which integrates Instrument Droid, Buzzer, and LED strip **(Completed Week 16)**

## Project Goals (Working Board)

- Power through 5V mini-USB (Input Power Jack replaced with 2-pin out to reduce confusion)
- Assemble the board without burning through entire supply of boards/parts (only 5 PCB boards)
- Arduino able to be programmed through ArduinoIDE using COM port (CH340g) via USB
- No significant power-rail collapse causing system failure
- No burning parts/shorts leading to burnt/non-functioning parts
- Support measurement of power supplies up to 12V, with peak current of 3A (momentarily)
- Reasonable ADC/DAC output (low-noise) with proper I2C communication
- Print measured Thevenin Resistance, Current, and Power Supply Voltage (Thevenin voltage, and No Load Voltage) to Serial Monitor
- Be on Schedule

## Sketch of the Schematic

### *Instrument Droid Connections*

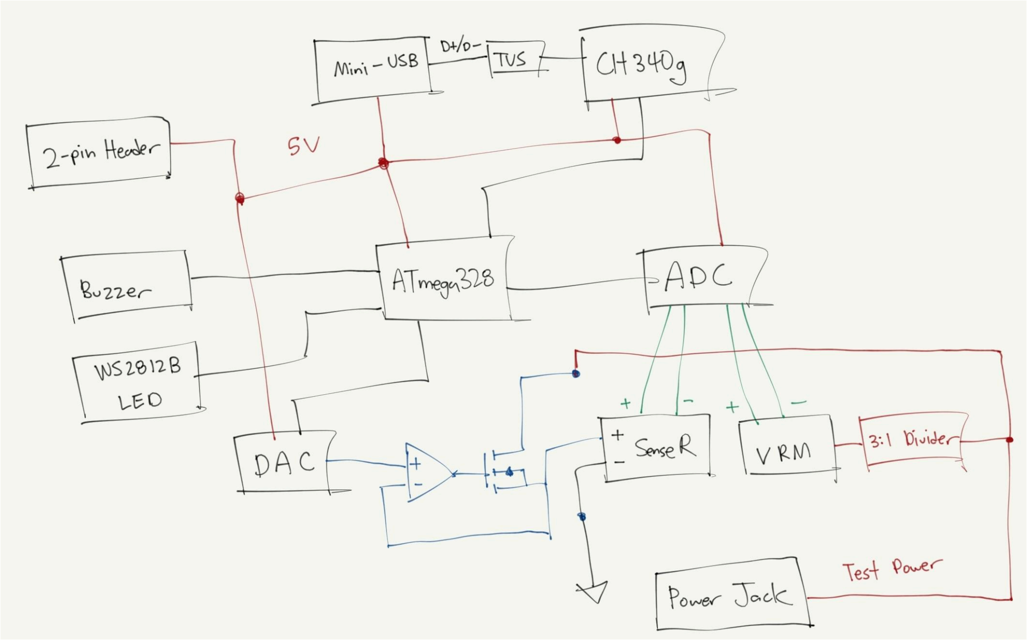


Figure 1: System Diagram for Instrument Droid

Circuit from the Golden Arduino board was reused, with minor changes. Micro-USB was changed to Mini-USB due to lack of parts in the lab. The device now only takes 5V power from USB input or 2-pin header. ADC (ADS1115), Op-Amp (MCP6002), DAC (MCP4725) were added for the Instrument Droid circuit. ADS1115 and MCP4725 communicate with the Atmega328 via I2C. Buzzer and 3 WS2812B LEDS are connected to Atmega328 through GPIO pins.

### *Instrument Droid Circuit*

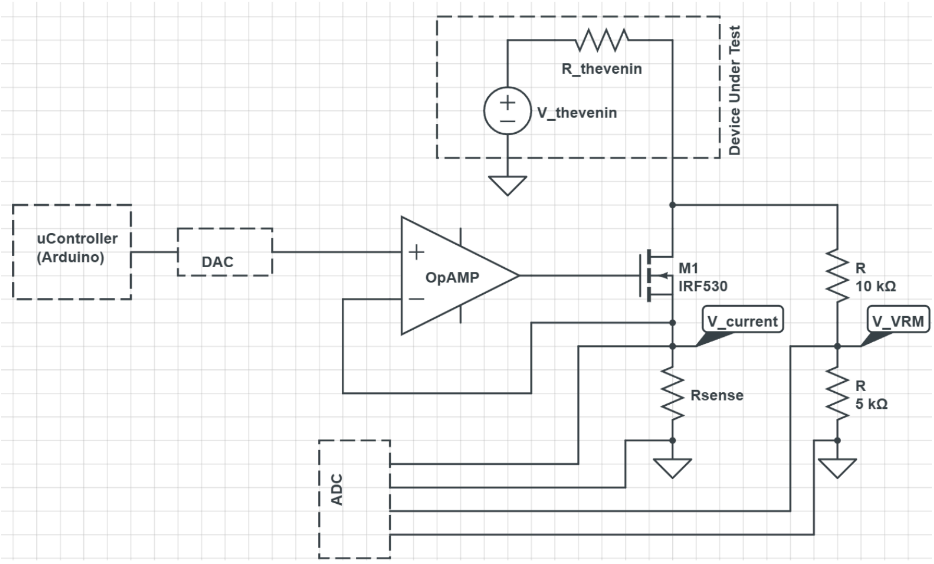
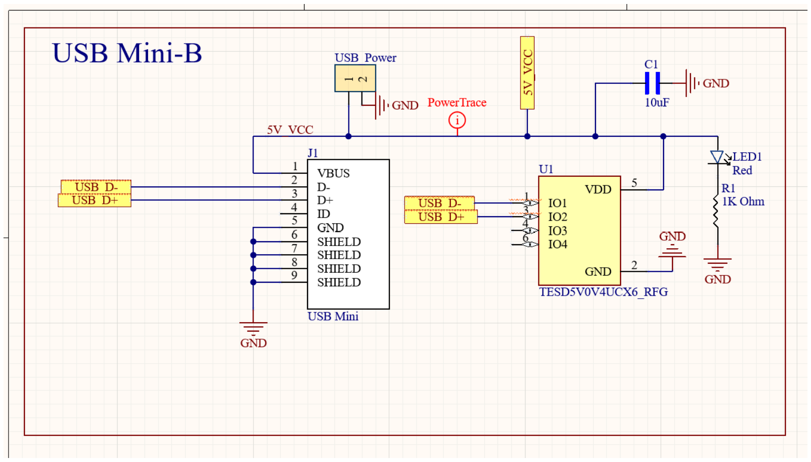
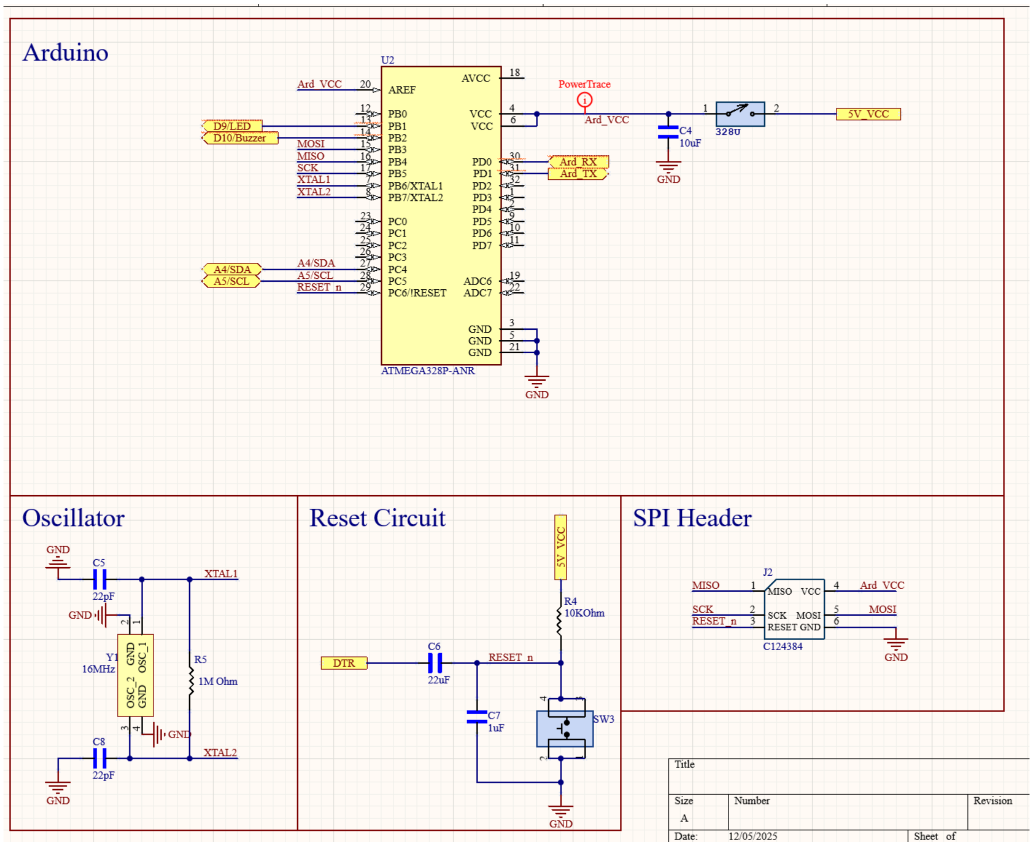


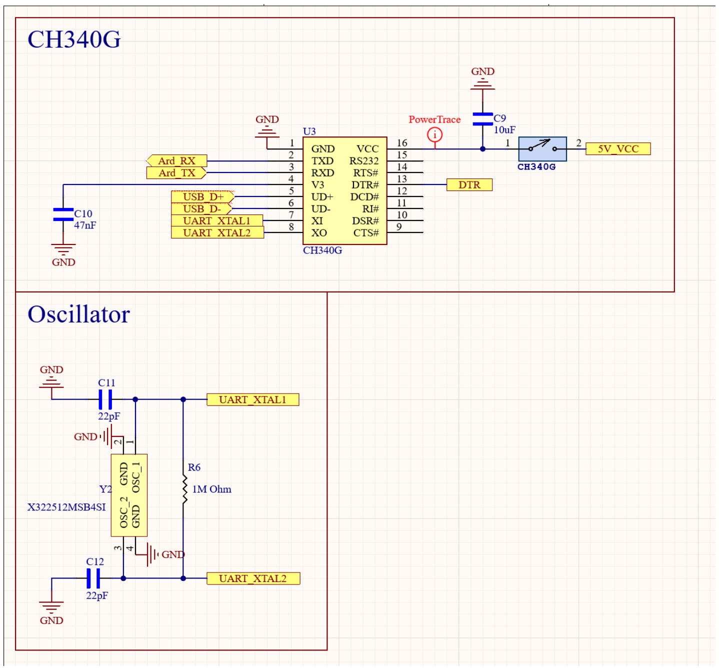
Figure 2: Schematic for Instrument Droid

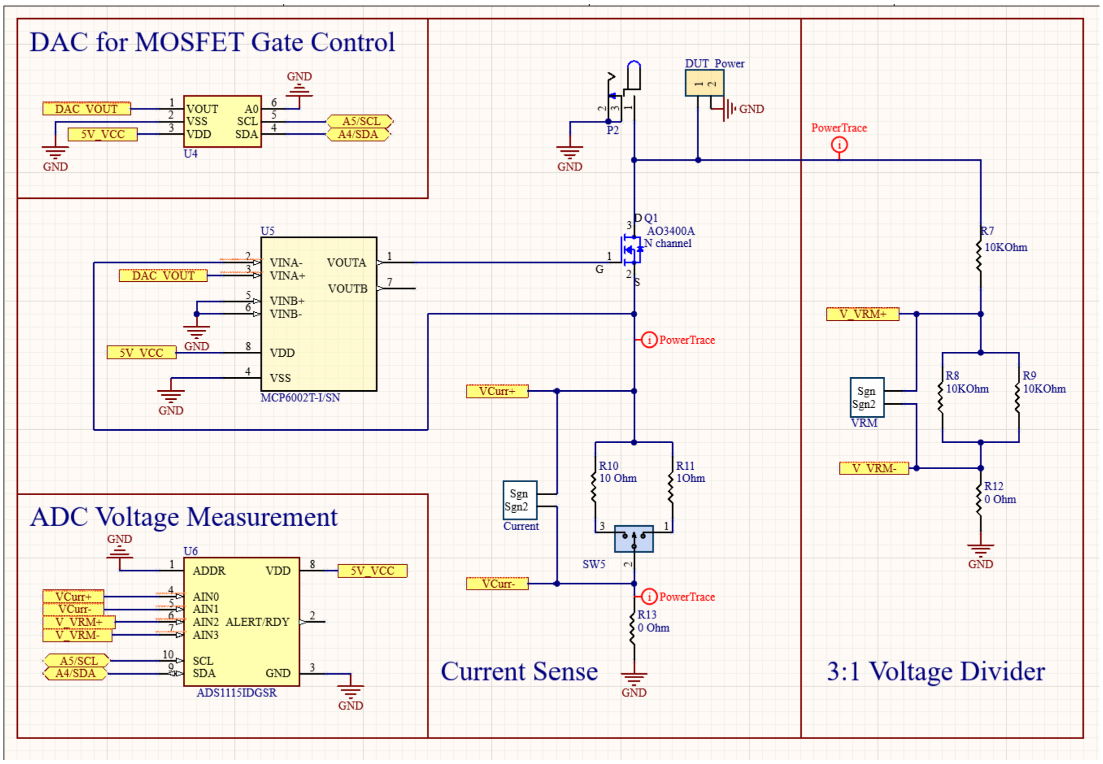
Arduino (Atmega328) sends commands to DAC to output specific voltage to the sense resistor. Feedback loop with Op-Amp and NMOS keeps the voltage stable when high current flows through the sense resistor. Voltage Regulator Module (VRM) is a voltage divider from external power supply under measurement. Both sense resistor and VRM are connected to ADC for differential voltage measurement.

## Altium Designer Schematic






CH340G
GND
C9
10uF
PowerTrace
GND
U3
1 GND VCC 16
2 TXD RS232 15
3 RXD RTS# 14
4 V3 DTR# 13
5 UD+ DCD# 12
6 UD- RI# 11
7 XI DSR# 10
8 XO CTS# 9
Ard_RX
Ard_TX
USB_D+
USB_D-
UART_XTAL1
UART_XTAL2
CH340G
1
2
CH340G
5V_VCC
DTR
C10
47nF
GND
Oscillator
GND
C11
22pF
UART_XTAL1
GND
Y2
X322512MSB4SI
GND OSC_1
OSC_2 GND
R6
1M Ohm
GND
C12
22pF
UART_XTAL2
GND


DAC for MOSFET Gate Control
GND
DAC_VOUT
5V_VCC
1 VOUT A0 6
2 VSS SCL 5
3 VDD SDA 4
U4
A5/SCL
A4/SDA
GND
P2
GND
DUT_Power
1 2
GND
PowerTrace
U5
2 VINA- VOUTA 1
3 VINA+
5 VINB+ VOUTB 7
6 VINB-
8 VDD
4 VSS
MCP6002T-I/SN
DAC_VOUT
GND
5V_VCC
GND
Q1
AO3400A
N channel
D
G
S
PowerTrace
R7
10KOhm
V_VRM+
VCurr+
Sgn
Sgn2
VRM
R8
10KOhm
R9
10KOhm
V_VRM-
R12
0 Ohm
GND
R10
10 Ohm
R11
1Ohm
Sgn
Sgn2
Current
SW5
VCurr-
PowerTrace
R13
0 Ohm
GND
ADC Voltage Measurement
GND
U6
1 ADDR VDD 8
4 AIN0
5 AIN1
6 AIN2 ALERT/RDY 2
7 AIN3
10 SCL
9 SDA GND 3
ADS1115IDGSR
5V_VCC
VCurr+
VCurr-
V_VRM+
V_VRM-
A5/SCL
A4/SDA
GND
Current Sense
3:1 Voltage Divider

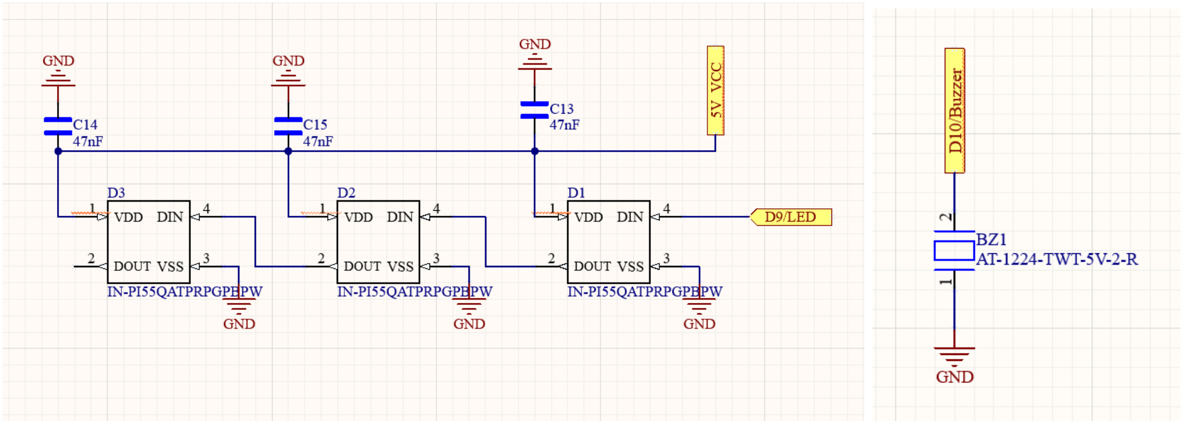


Figure 3: Altium Schematic for Board 4 (From Top to Bottom: Mini-USB, Atmega328, CH340g, Instrument Droid, LED/Buzzer)

Many components from Golden Arduino (Board 3) were removed to make the PCB size smaller. Standard Arduino Header pins were removed, along with the majority of test points. Power jack input was removed, alongside 3.3V LDO. New components were added for the Instrument Droid functionality.

## PCB Board Layout

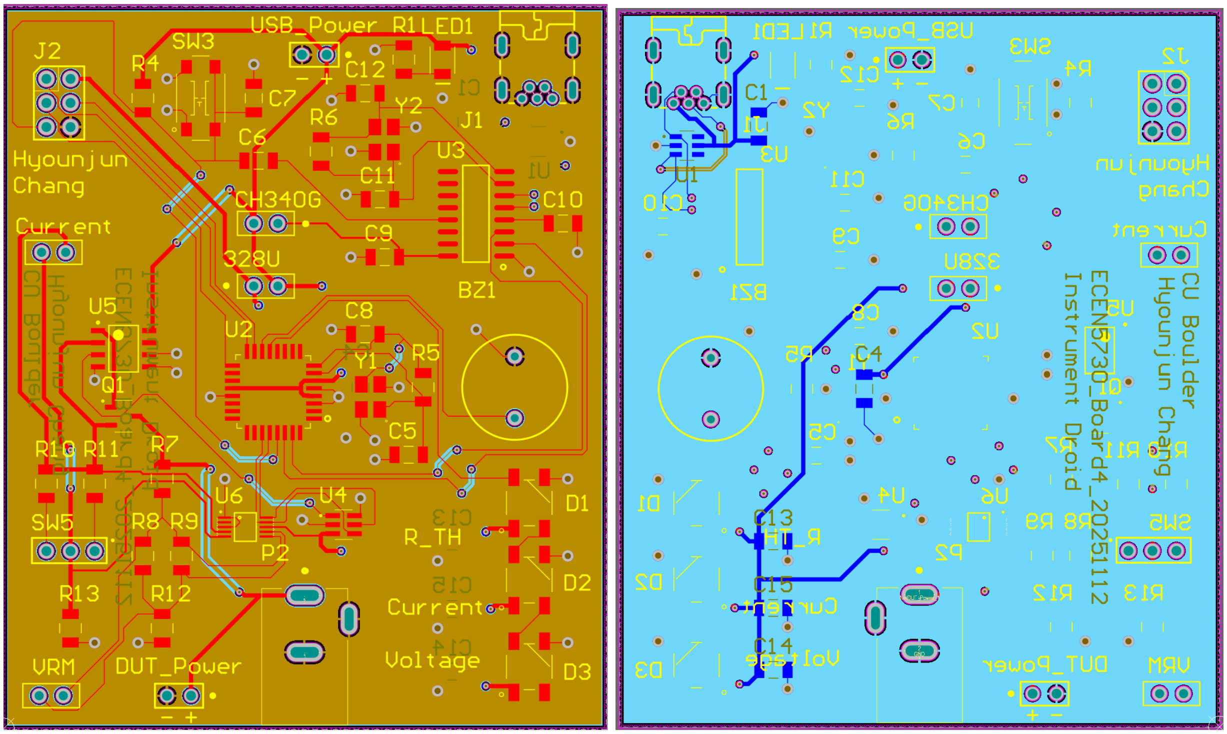


Figure 4: PCB Design on Altium Designer 2D Front View (Left), 2D Back View (right)

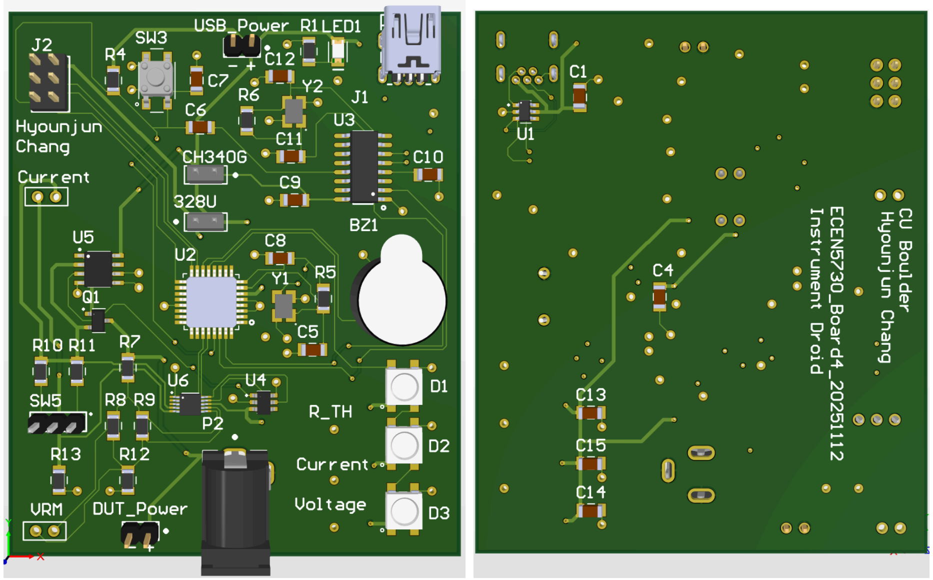


Figure 5: PCB Design on Altium Designer 3D Front View (Left), 3D Back View (right)

Dimension of the PCB board is 2500mil by 3000mil, which is smaller than the Golden Arduino Board (Board 3). Header pins from Golden Arduino were removed except the 6-pin header (SPI) required to burn the bootloader. The PCB is 4-layer, with Layer 2 and 3 being ground planes. Some decoupling capacitor and TVS circuit was placed in the back layer. Routing was primarily done on Layer 1 and Layer 4, with through-hole vias. Mini-USB is placed on the top right, and Power Jack for power supply testing is placed on the bottom. Being able to place parts in the back also contributed to small board size.

## Picture of Physical Board

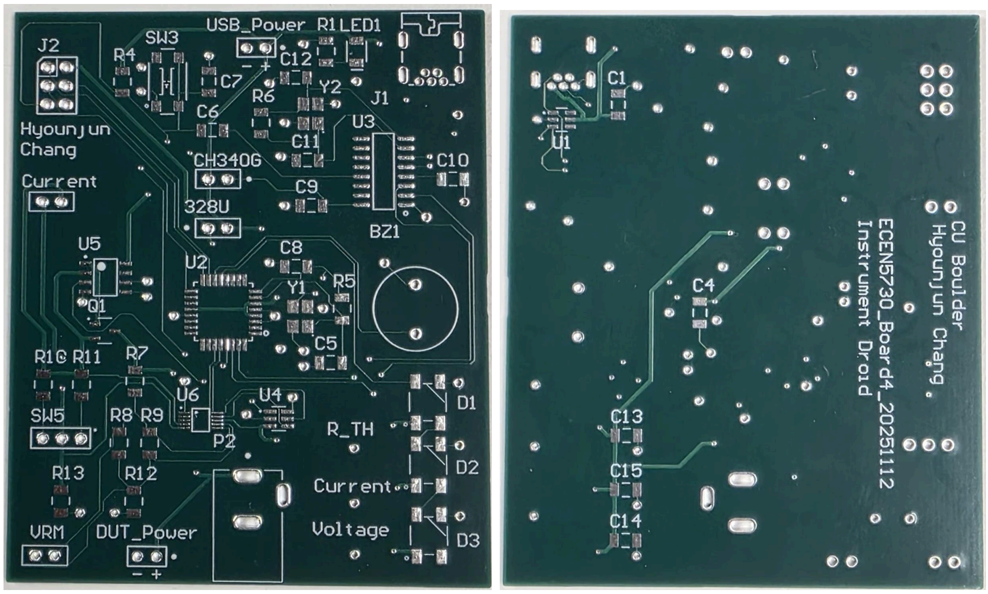


Figure 6: Unassembled Instrument Droid PCB (Left: Front, Right: Back)

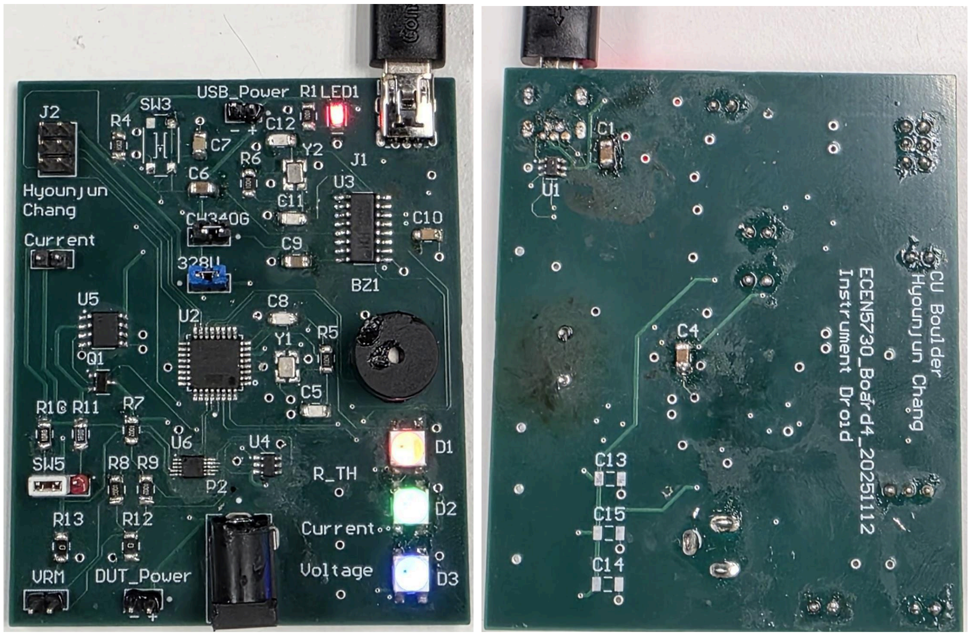


Figure 7: Assembled Instrument Droid PCB (Left: Front, Right: Back)

Not all parts were placed onto the assembled board. Pressing SW3 (Reset Switch) caused the CH340g to brick, so the switch was removed to prevent future damages. Three decoupling capacitors for WS2812B LEDs were not placed since they were not deemed necessary to function properly.

# Board Functionality:

Test Points:
USB_Power: 5V USB Power Input/Test Point
Current: Sense Resistor Test Point
VMRM: Voltage Regulator Module Test Point
DUT_Power: Measurement Power Supply Input/Test Point

## Expected Working Board:

- Stable 5V power-rail output from USB, with indicator LED turning on
- CH340g functional, able to communicate with laptop via COM port for communication
- Arduino programmable via COM port after burning bootloader
- ATmega328 able to get differential voltage readings from ADC (ADS1115) for sense resistor and VRM
- ATmega328 able to control output voltage through current resistor through DAC (MCP4725)
- Voltage from DAC (MCP4725) stable throughout its duration regardless of high current
- After programmed with custom program: makes a buzzer sound when it starts power supply measurements
- Separate power input from power jack (only for measurements), no power

- LED strips able to be programmed individually (red, green, and blue respectively for the custom program)
- DAC (MCP4725) is able to set desired voltage levels, in 100ms with increments over intervals
- Selectable sense resistor of 10 ohms or 1 ohm
- Reasonable ADC readings/DAC output, with acceptable noise levels
- Relatively accurate calculation of Thevenin Resistance based upon sensor readings
- No burning parts or shorts

## Actual Assembled Board:

### *USB Power*

Indicator light was placed right next to USB, and only powers on when the USB is plugged in. The power jack input for measurement does not power the system at all.

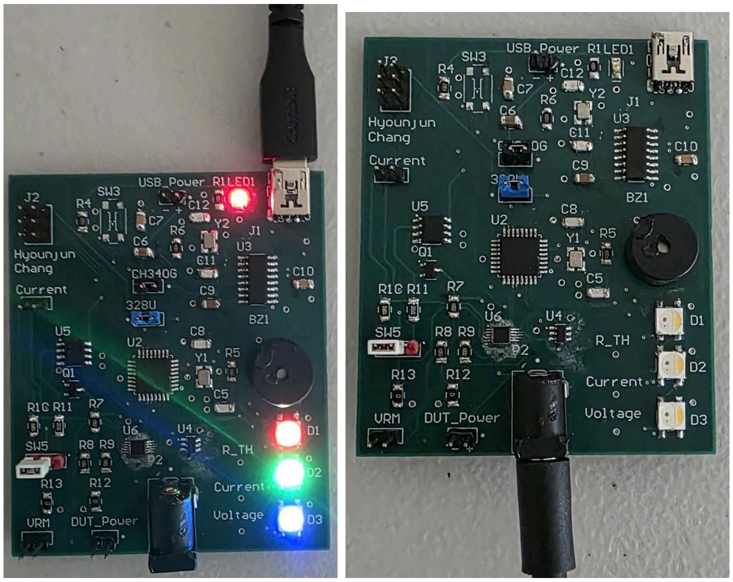
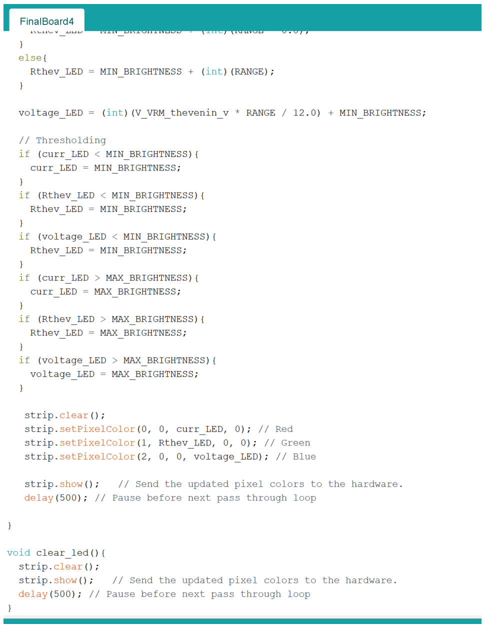


Figure 8: Instrument Droid with only USB power plugged-in (left), only Power Jack plugged-in (right)

### *WS2812B Strip*

When measurement is taking place, the 3 LEDS (D1, D2, D3) are lit in red, green, and blue as seen in Figure 8. Brightness of the LEDs are controlled by Thevenin Resistance, Current, and supply voltage. However, the brightness controls are not that great as even low value creates a bright light. The value is set to 20 (max. 256) even with minimum readings, which is still bright.



```
  }
  else{
    Rthev_LED = MIN_BRIGHTNESS + (int)(RANGE);
  }

  voltage_LED = (int)(V_VRM_thevenin_v * RANGE / 12.0) + MIN_BRIGHTNESS;

  // Thresholding
  if (curr_LED < MIN_BRIGHTNESS){
    curr_LED = MIN_BRIGHTNESS;
  }
  if (Rthev_LED < MIN_BRIGHTNESS){
    Rthev_LED = MIN_BRIGHTNESS;
  }
  if (voltage_LED < MIN_BRIGHTNESS){
    Rthev_LED = MIN_BRIGHTNESS;
  }
  if (curr_LED > MAX_BRIGHTNESS){
    curr_LED = MAX_BRIGHTNESS;
  }
  if (Rthev_LED > MAX_BRIGHTNESS){
    Rthev_LED = MAX_BRIGHTNESS;
  }
  if (voltage_LED > MAX_BRIGHTNESS){
    voltage_LED = MAX_BRIGHTNESS;
  }

   strip.clear();
   strip.setPixelColor(0, 0, curr_LED, 0); // Red
   strip.setPixelColor(1, Rthev_LED, 0, 0); // Green
   strip.setPixelColor(2, 0, 0, voltage_LED); // Blue

   strip.show();   // Send the updated pixel colors to the hardware.
   delay(500); // Pause before next pass through loop

}

void clear_led(){
  strip.clear();
  strip.show();   // Send the updated pixel colors to the hardware.
  delay(500); // Pause before next pass through loop
}
```

Figure 9: Micro code responsible for controlling WS2812B LEDs. LEDs are colored Red, Green, and Blue respectively.

### *CH340g and Programming Arduino*

CH340g was recognized by a laptop and the Instrument droid was able to be programmed through ArduinoIDE after the bootloader was burned via its 6-pin header on top left.

Figure 10: CH340g recognized in COM port (Left), Successful Arduino programming (Right)

### *Reset Circuit*

The reset circuit, which is connected to the DTR pin on CH340g has hard faults. A button press on the reset switch caused the CH340g to fry, making it non-functional. The reset switch was removed to prevent future damages.

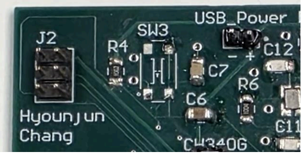


Figure 11: Reset Switch (SW3) left as a no connect post-assembly

## *Selectable Sense Resistor*

A jumper can be used in a 3-pin connector to select a 10-ohm or 1-ohm resistor. For high voltage test inputs, 1-ohm resistor selection is not recommended.

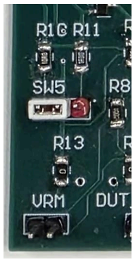


Figure 12: Sense Resistor selected for 10-ohm with a jumper in SW3

## *I2C Bus*

There is ongoing data transfer between ATmega328 and the 2 I2C devices (ADC and DAC). There are no test points for SDA or SCL, so a probe was placed next to the pins. SCL was not able to be captured due to issues with capacitance, but SDA transfer is happening.

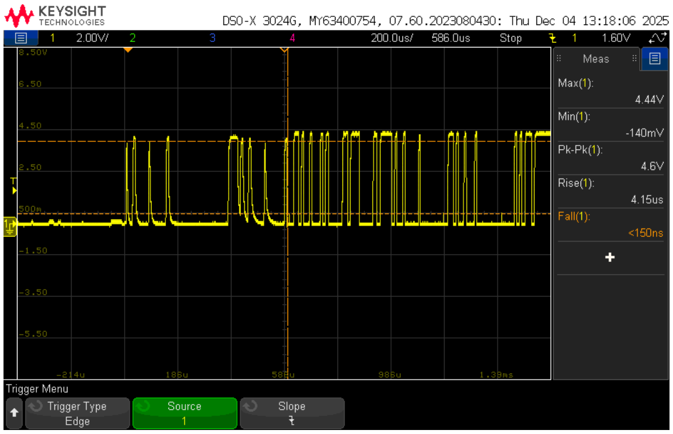


Figure 13: SDA Signal from ATmega328

## *DAC and Sense Resistor Current*

DAC is able to control voltages on the sense resistor by controlling the voltage output of the positive end of the sense resistor. Voltage is stable through the 100ms measurement duration. Rise time of the edge is slow at around 3us, and no large overshoots were found.

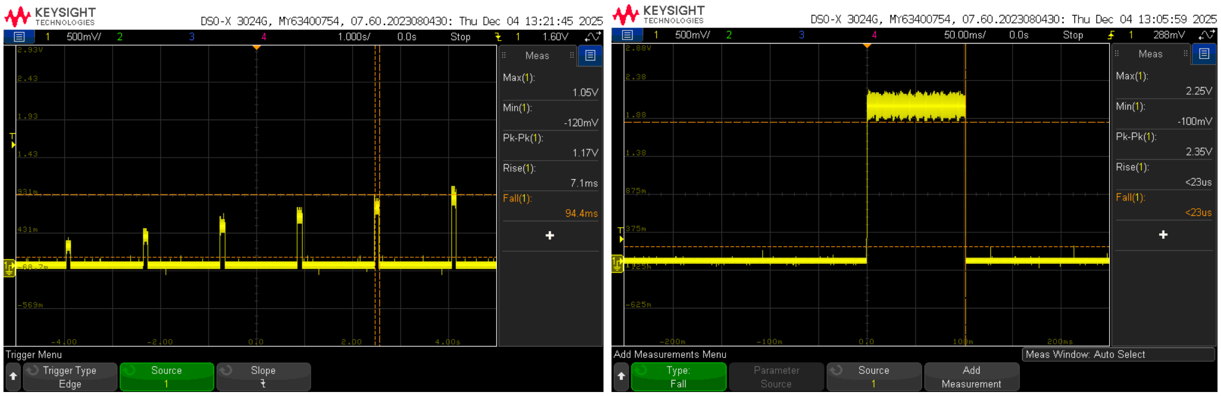


Figure 13: Gradually increasing MCP4725 (DAC) output (Left), zoomed in at 2.25V (Right)

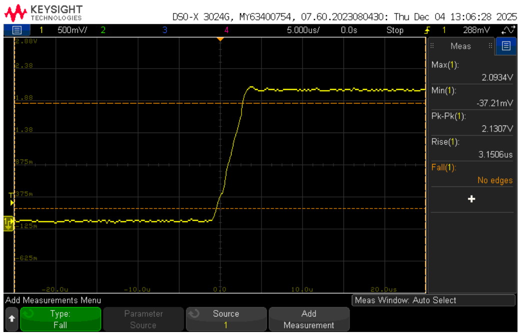


Figure 14: Rising edge of MCP4725 output at 2.25V

### *Buzzer Output*

Buzzer creates a sound at different frequencies (gradually higher pitch tones) when measurement starts. Frequency changes are visible on the oscilloscope. There are some high overshoots, with peak-to-peak voltage reaching 7.28V, a result of lack of a small capacitor.

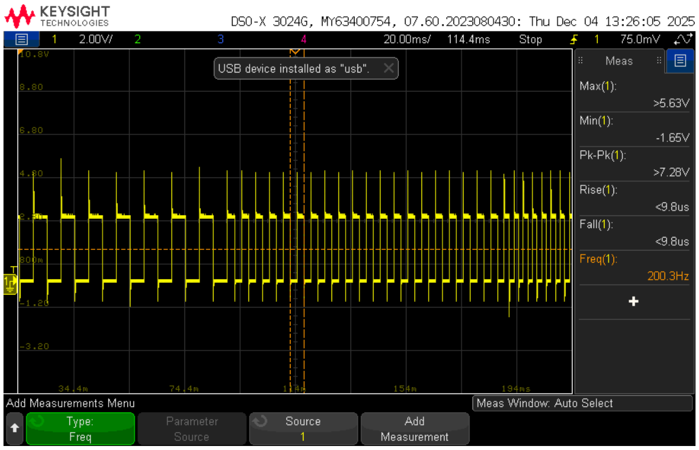


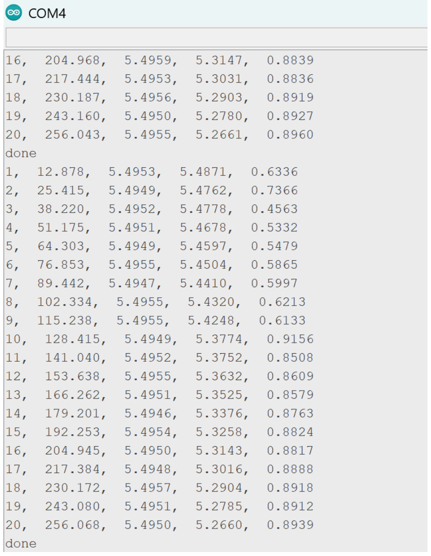
Figure 15: Buzzer output at the start of measurement

### *ADC Readings and Serial Monitor Output*

Reasonable ADC readings are sent over the I2C bus and printed to the serial console. Two differential signals are measured (Sense Resistor, VRM), and a reasonable output is shown from a 5V power supply, indicating that the noise level is acceptable. The calculated values are not too precise at low thevenin resistance values due to precision and noise levels during measurement.

The results are printed in following order:

1. **Iteration**
2. **Sense Resistor Current in mV)**
3. **Thevenin Voltage**
4. **Load Voltage**
5. **Thevenin Resistance**

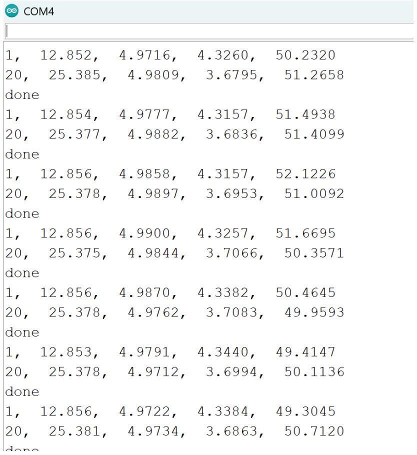


```
16,  204.968,  5.4959,  5.3147,  0.8839
17,  217.444,  5.4953,  5.3031,  0.8836
18,  230.187,  5.4956,  5.2903,  0.8919
19,  243.160,  5.4950,  5.2780,  0.8927
20,  256.043,  5.4955,  5.2661,  0.8960
done
1,  12.878,  5.4953,  5.4871,  0.6336
2,  25.415,  5.4949,  5.4762,  0.7366
3,  38.220,  5.4952,  5.4778,  0.4563
4,  51.175,  5.4951,  5.4678,  0.5332
5,  64.303,  5.4949,  5.4597,  0.5479
6,  76.853,  5.4955,  5.4504,  0.5865
7,  89.442,  5.4947,  5.4410,  0.5997
8,  102.334,  5.4955,  5.4320,  0.6213
9,  115.238,  5.4955,  5.4248,  0.6133
10,  128.415,  5.4949,  5.3774,  0.9156
11,  141.040,  5.4952,  5.3752,  0.8508
12,  153.638,  5.4955,  5.3632,  0.8609
13,  166.262,  5.4951,  5.3525,  0.8579
14,  179.201,  5.4946,  5.3376,  0.8763
15,  192.253,  5.4954,  5.3258,  0.8824
16,  204.945,  5.4950,  5.3143,  0.8817
17,  217.384,  5.4948,  5.3016,  0.8888
18,  230.172,  5.4957,  5.2904,  0.8918
19,  243.080,  5.4951,  5.2785,  0.8912
20,  256.068,  5.4950,  5.2660,  0.8939
done
```

Figure 16: Serial Monitor Output for 5V External Power Supply

### *Thevenin Resistance Measurement with a Known Source*

Errors in measurement with a known 50-ohm Thevenin Resistance source (Function Generator) is relatively low, indicating that measurement values are relatively accurate. Calculated Thevenin resistance values range from 49-ohms to 52-ohms.



```
1,  12.852,  4.9716,  4.3260,  50.2320
20,  25.385,  4.9809,  3.6795,  51.2658
done
1,  12.854,  4.9777,  4.3157,  51.4938
20,  25.377,  4.9882,  3.6836,  51.4099
done
1,  12.856,  4.9858,  4.3157,  52.1226
20,  25.378,  4.9897,  3.6953,  51.0092
done
1,  12.856,  4.9900,  4.3257,  51.6695
20,  25.375,  4.9844,  3.7066,  50.3571
done
1,  12.856,  4.9870,  4.3382,  50.4645
20,  25.378,  4.9762,  3.7083,  49.9593
done
1,  12.853,  4.9791,  4.3440,  49.4147
20,  25.378,  4.9712,  3.6994,  50.1136
done
1,  12.856,  4.9722,  4.3384,  49.3045
20,  25.381,  4.9734,  3.6863,  50.7120
done
```

Figure 17: Serial Monitor Output for Function Generator, +5V bias, 30mVpp, 0.5Hz

*Video Demo*

A video of working Instrument Droid can be found here.

## Conclusion:

Designing systems on PCB creates many challenges; from part selection, routing, signal integrity, to assembly, many steps can cause problems, and finding ways to reduce risks is crucial. Reusing previous designs and parts tested to work significantly saves time required for testing, and therefore reduces chances of hard and soft errors during bring-up.

Using higher-count PCB design increases costs, but creates much more options for efficient routing. Placing components in the back can reduce board size, but should be carefully considered since placing thick parts on the back layer may lead to mechanical problems.

Embedded software should be considered when designing PCB hardware as well, and expected behavior during bring-up should include testing with software. Bad software practices, such as letting the sense resistor run too long can cause problems to hardware.

There will be issues when building and testing a system, and good debugging skills, placement of test points, and PCB design practices will help you make modifications in the future to improve future designs.

# Project Analysis

There were many errors in the board in the circuit design, PCB layout, and assembly.

**Hard Errors:**

- Pressing the Reset Button led to a short in CH340g, damaging the IC. Same behavior occurred in board 3 when wrong size capacitors were placed. The switch was removed to prevent future damages, and the CH340g was replaced.
- Initial misassembly of the board's 10-pin ADC (ADS1115) led to high current being consistently sent through the 10-ohm sense resistor, burning the ADC and sense resistor.
- Circuit may be vulnerable at high currents, as indicated by burning of 10-ohm sense resistor suggests that 1-ohm resistor may not be safe to use

**Soft Errors:**

- Extra parts were required during assembly (CH340g, ADS1115) due to circuits malfunctioning with improper contacts to the solder mask.
- Absence of decoupling capacitors for ADC and DAC, which was lost during Design Review
- Incorrect software reading wrong pins and making wrong calculations increased debugging time
- Absence of I2C bus headers for debugging
- DNP of Reset Switch after hard error

## What went well:

- Better assembly of the board: better techniques led to no boards being burnt
- Reuse of design from Golden Arduino board significantly reduced PCB design time
- Easy to understand serial monitor output from software
- Reducing test points for parts tested during bring-up of Golden Arduino (Board 3)
- Testing of new parts (Buzzer, SmartLED) with microCode, separate from functionality of instrument droid
- Debugging: faulty parts were quickly identified and replaced, a result of familiarity of the design from Board 3
- Board size: reduced to 2500mils to 3000mils, which is smaller than Golden Arduino despite having additional IC.
- No issues with 4-layer PCB design, with functioning parts placed on back layer

## What didn't go well:

- Reset circuit and its decoupling capacitors were not properly placed, leading to hard failure, and one feature being removed (Reset Circuit)
- Disorganization of software code led to longer debugging times of ADC - despite the hardware being perfectly fine
- PCB design connected to different pins from existing micro-code from testing the parts, and this was not recorded
- Lack of thorough testing of higher voltage power supplies and 1-ohm resistor; after experiencing a damage in ADC and sense resistor, only supplies close to 5V were tested
- No alternative methods to get Thevenin resistance aside from USB port
- Risk of blowing up a USB port at high currents: to avoid this only supplies close to 5V were tested, so not all design specs were tested.
- Carefully observing new parts: buzzer was placed with incorrect polarity, so it had to be re-soldered.

## Future Improvements:

- Devise a way to verify functionality of Instrument droid with higher voltage supplies without potentially damaging a USB port
- Better measurement techniques to prevent part blow-ups during testing, especially with high-current
- Smaller, dense PCB routing: majority of layer 4 is unused, and not many parts are placed in the back. Board can be made even smaller with through planning
- Designing boards with future expandability: no I2C pins were placed into the instrument droid. Placement of SDL and SDA pins would have allowed expansion of features such as a I2C
- Proper reset circuit with smaller capacitors to reduce likelihood of excess current going to CH340g through DTR pin
- Better use of DNP in case testing goes wrong (ex. additional slots for decoupling capacitors to test different sizes or more expanded slots for LED)
- Better functionality on the software with better brightness controls