Lab 3: A Golden Arduino PCB

Objective:

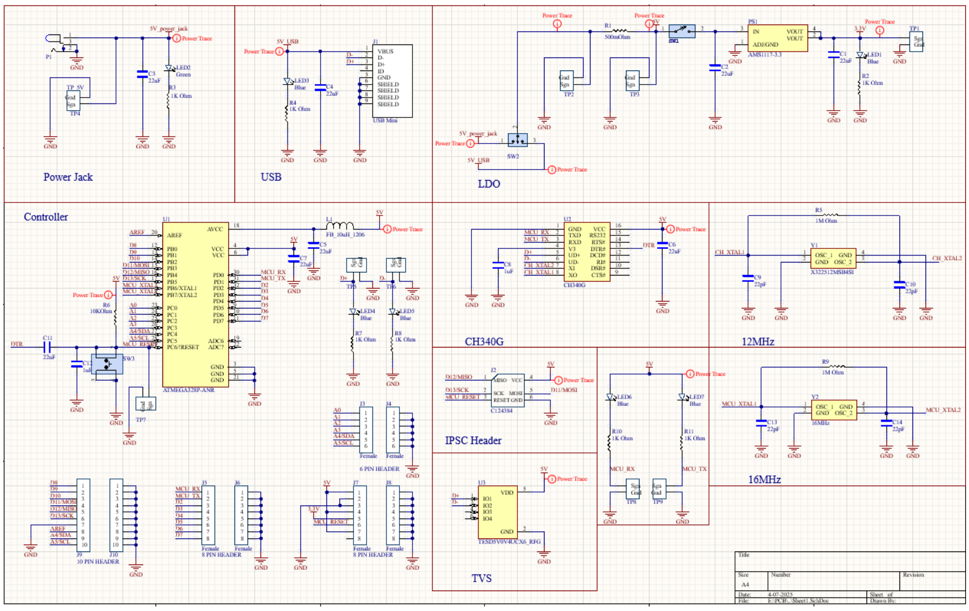
To design and prototype a fully functional Arduino Uno board using the complete design process, applying best practices in signal routing, noise reduction, and board bring-up. The project aims to enhance testability, assembly, and debugging by integrating features such as test points, indicator LEDs, and isolation switches. Final validation includes comparing the custom board's noise and emissions metrics to those of a commercial board, demonstrating the effectiveness of thoughtful engineering design.

Bill of Material:

	A	B	C	D
1	Name	Description	Designator	Quantity
2				
3	22uF	22uF ±10% 25V X5R 1206 Multilayer Ceramic Capacitors MLCC - SMD/SMT RoHS	C1, C2, C3, C4, C5, C6, C7, C11	8
4	1uF	MULTILAYER CERAMIC CAPACITORS MLCC - SMD/SMT 1UF 50V 1206 ROHS	C8, C12	2
5	22pF	22pF ±5% 1kV C0G 1206 Multilayer Ceramic Capacitors MLCC - SMD/SMT RoHS	C9, C10, C13, C14	4
6	USB Mini	Through Hole USB Connectors RoHS	J1	1
7	NRPN032PAEN-RC	Connector Header Through Hole 6 position 0.079" (2.00mm)	J2	1
8		6 Position Header Connector 0.100" (2.54mm) Through Hole Tin	J3, J4	2
9		8 Position Header Connector 0.100" (2.54mm) Through Hole Tin	J5, J6, J7, J8	4
10	PPTC101LFBN-RC	10 Position Header Connector 0.100" (2.54mm) Through Hole Tin	J9, J10	2
11	FB_10uH_1206	25mA 10uH ±10% 800mΩ 1206 Inductors (SMD) ROHS	L1	1
12	Blue	blue 1206 Light Emitting Diodes (LED) ROHS	LED1, LED3, LED4, LED5, LED6, LEI	6
13	Green	Green 513~528nm 1206 Light Emitting Diodes (LED) RoHS	LED2	1
14	Power Jack	Power Barrel Connector Jack 2.10mm ID (0.083"), 5.50mm OD (0.217") Through Hole, Right Angle	P1	1
15	AMS1117-3.3	LOW DROPOUT REGULATORS(LDO) POSITIVE FIXED 1.3V @ 800MA 15V 3.3V 1A SOT-223 RO	PS1	1
16	500m	Resistors/Shunt 250mW 500mΩ ±800ppm/? ±5% 1206 Current Sense Resistors ROHS	R1	1
17	1k	CHIP RESISTOR - SURFACE MOUNT 1KOHMS ±1% 1/4W 1206 ROHS	R2, R3, R4, R7, R8, R10, R11	7
18	1M	CHIP RESISTOR - SURFACE MOUNT 1MOHMS ±1% 1/4W 1206 ROHS	R5, R9	2
19	10k	10 kOhms ±5% 0.25W, 1/4W Chip Resistor 1206 (3216 Metric) Automotive AEC-Q200 Thick Film	R6	1
20	SW_2Pin_100mil_Switch	2Pin Header	SW1	1
21	PREC003SAAN-RC	Connector Header Through Hole 3 position 0.100" (2.54mm)	SW2	1
22	TL3305AF160QG	Tactile Switch SPST-NO Top Actuated Surface Mount	SW3	1
23		Test Point 300 mil centers	TP1, TP2, TP3	3
24	TP_5V	Test Point 300 mil centers	TP4	1
25	10x Probe TP	Test Point 300 mil centers	TP5, TP6, TP7, TP8, TP9	5
26	ATMEGA328P-ANR	AVR AVR® ATmega Microcontroller IC 8-Bit 20MHz 32KB (16K x 16) FLASH 32-TQFP (7x7)	U1	1
27	CH340G	Transceiver USB 2.0 2Mbps SOP-16_150mil USB ICs RoHS	U2	1
28	TESD5V0V4UCX6_RFG	SOT-26, 5V, 75W, 0.6pF, ESD Protection	U3	1
29	X322512MSB4SI	SMD Crystal Resonators 12MHz ±10ppm SMD-3225_4P RoHS	Y1	1
30	X322516MLB4SI	±10ppm 16000000Hz -40?~85? ?????? 60Ω 9pF SMD3225-4P Crystals ROHS	Y2	1

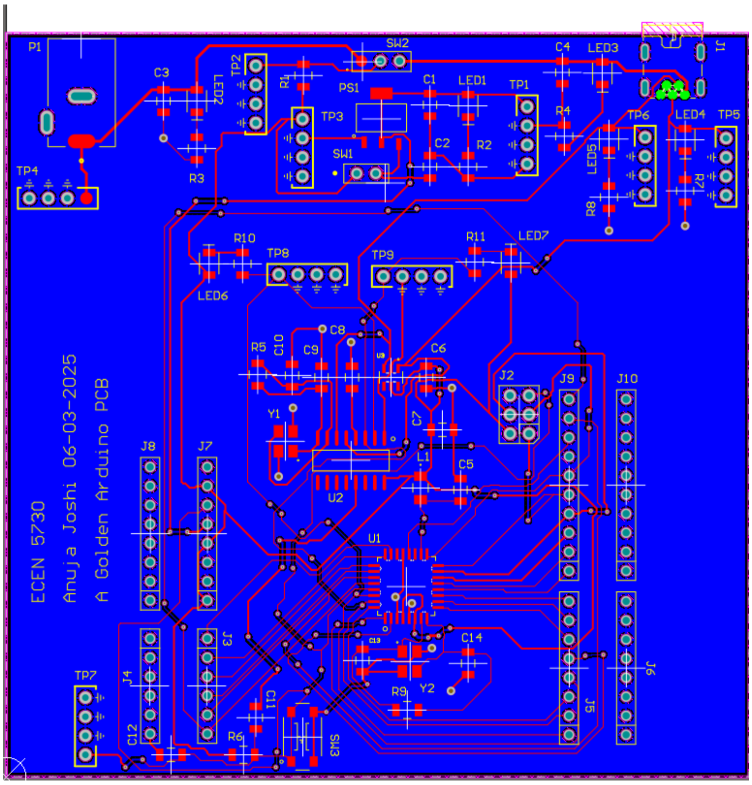
Altium Schematic:

The schematic illustrates the complete design of the custom Arduino-compatible board. It includes the Atmega328 microcontroller, USB-to-UART interface (CH340G), power input via DC jack and USB, a 3.3V LDO regulator, and associated peripherals such as the crystal oscillator, reset circuitry, and indicator LEDs. Key test points are added for power rails, communication lines, and inrush current measurement. Careful consideration has been given to signal integrity and layout practices, with provisions for debugging and measurement during board bring-up.



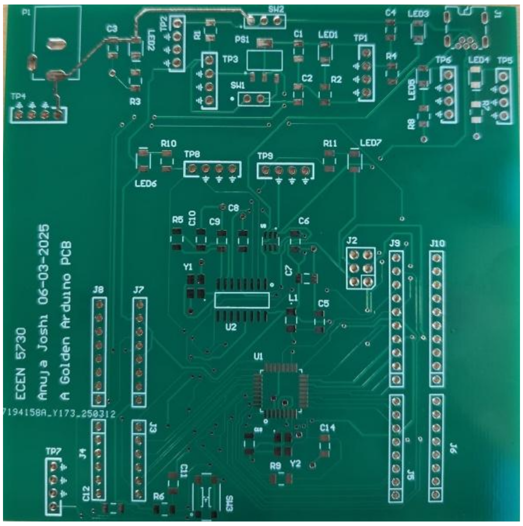
Board Layout:

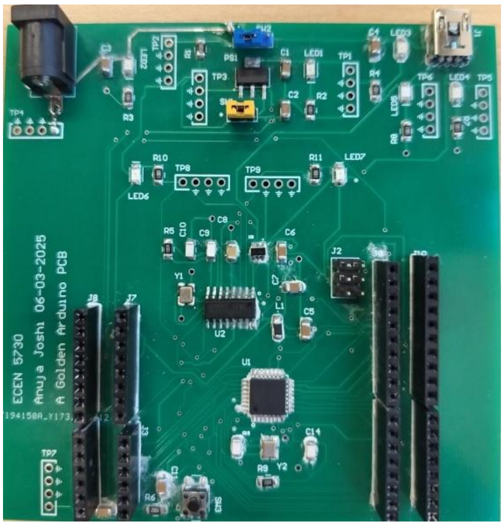
The board layout is optimized for compact component placement while ensuring signal integrity and minimizing noise. Oscillators are positioned close to the microcontroller to reduce clock jitter, and the RX/TX lines are routed with matched lengths to maintain signal timing and minimize reflection. This careful layout contributes to stable operation and reliable communication.



Board Pictures:

The board was assembled using a combination of hand-soldering. Minor adjustments, like bending and manually soldering the 5V pin of the power jack due to a footprint issue, were made during assembly. The board was cleaned post-soldering to remove flux residues and ensure stable operation.





Before assembly After assembly

Plan of Record:

1. Power System:
A power input plug compatible with standard "AC to 5V DC adaptors" will supply primary power to the board. A switching mechanism will be included to toggle between USB and external 5V power sources. A 3.3V Low Dropout Regulator (LDO) will also be added to provide regulated 3.3V output via header pins for peripheral support.

2. USB Interface & Protection:
A USB mini-B connector will enable PC interfacing for programming and serial communication. To protect against voltage spikes, a TVS (Transient Voltage Suppression) diode will be placed on the USB data lines. A CH340G USB-to-Serial chip will provide UART communication between the PC and the microcontroller.

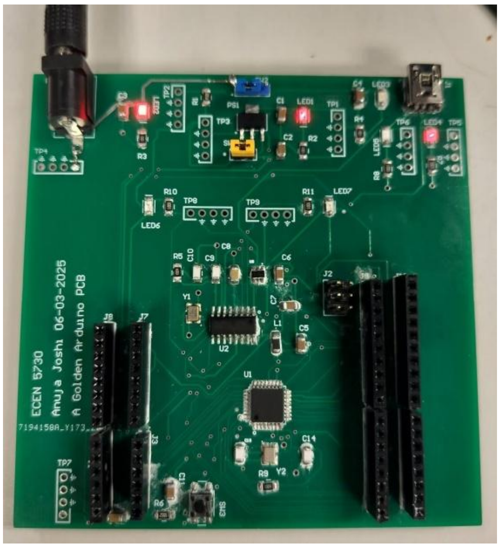
3. Microcontroller & Clocking:
The core of the board will be the Atmega328 microcontroller, which will be bootloaded via an In-Circuit Serial Programming (ICSP) header. A 16 MHz crystal oscillator will be used to clock the Atmega328, while a 12 MHz crystal will support the CH340G for stable USB communication.

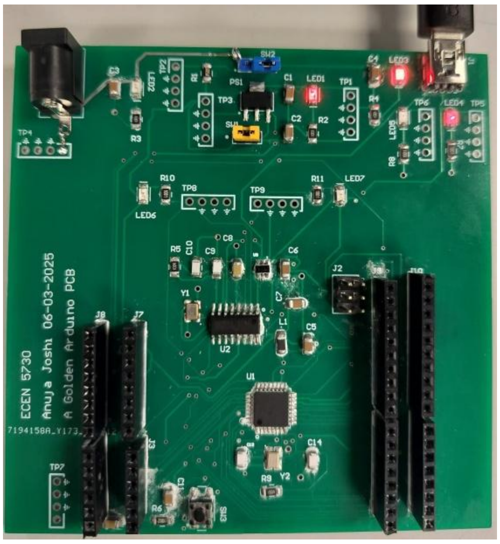
4. Inrush current:
A 500mΩ sense resistor will be added in series with the power rail, with an adjacent test point to enable differential measurement of inrush and steady-state current using a scope or differential voltmeter.

5. Debugging :
The board will include indicator LEDs for power status (main 5V supply, 3.3V LDO, and USB). A reset circuit will allow for manual board reset during development. For visibility and signal probing, test points will be added on critical signals including the 5V rail, 3.3V, and USB D+/D- lines. Additionally, isolation headers will be placed for the Atmega328 to aid in debugging and partial subsystem testing.

<u>Board bring up:</u>

1. Power Indicators:
Upon powering via both USB and DC jack, the respective indicator LEDs turn on, confirming power distribution and correct polarity.

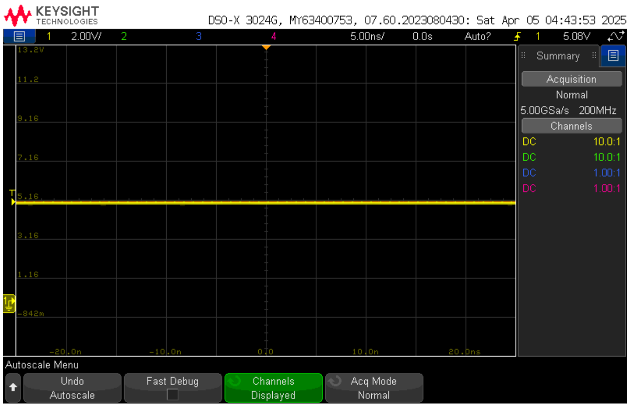




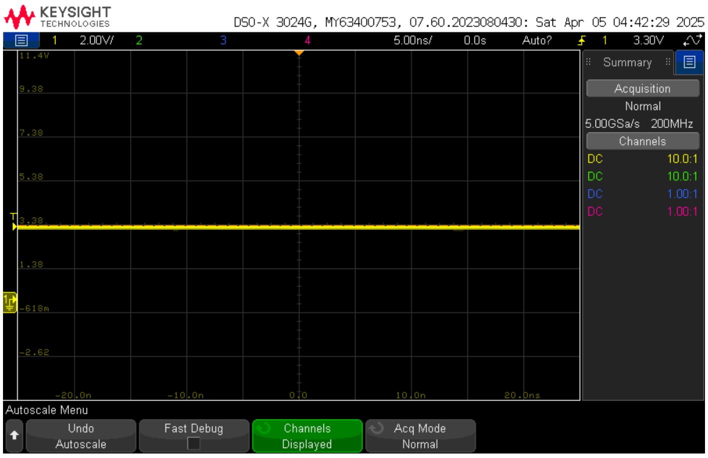
Power on using power jack

Power on using USB

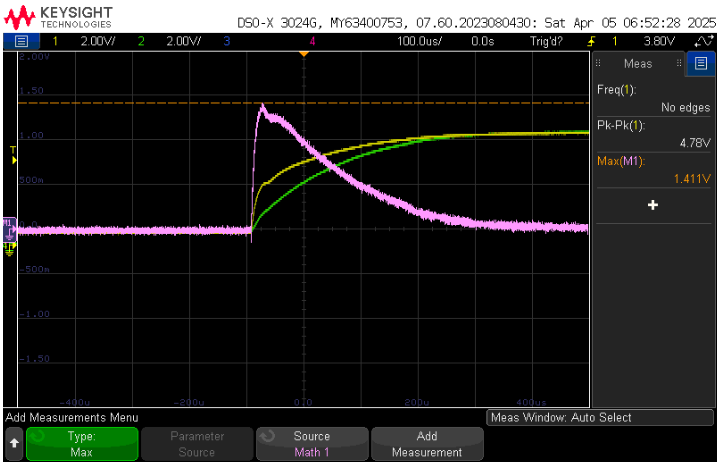
2. Power Source Switching:
The switch correctly toggles between USB and power (jack), with seamless power transition observed and no voltage dips.



3. 3.3V LDO Output Verification:
A stable 3.3V output was observed on the corresponding test point, confirming proper operation of the LDO regulator.



4. Inrush Current Measurement:
Inrush and steady-state current were measured across the 500mΩ sense resistor using differential probes. A brief inrush spike was observed at power-up, followed by a stable operating current.



Calculations:

Vpp= 1.411V

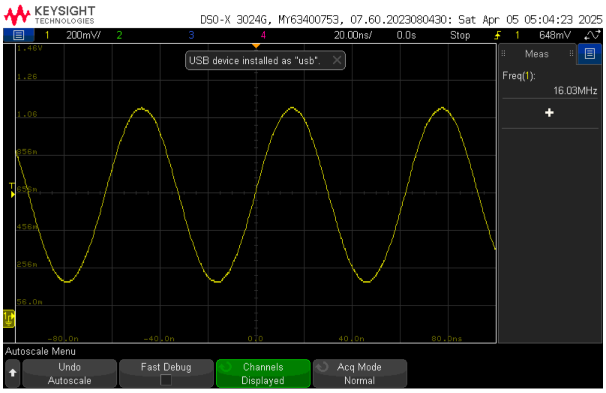
Resistor = 500mV

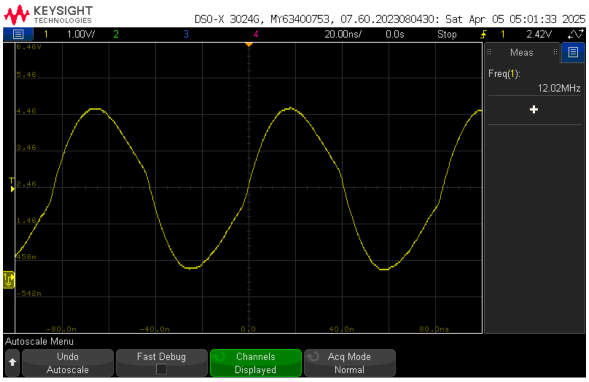
Inrush current = 1.411/0.5

=2.822A

5. Bootloading the Atmega328:
Successfully bootloaded the Atmega328 using a commercial Arduino Uno configured as ISP. Bootloader verified via Arduino IDE detection and sketch upload.

Verified Bootloading by verifying crystal frequency of Atmega328(16MHz) crystal and CH340G(12MHz)

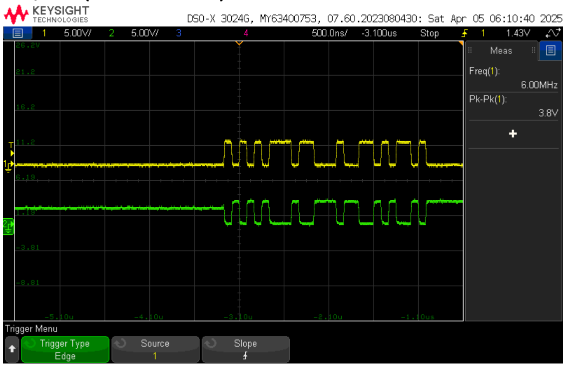


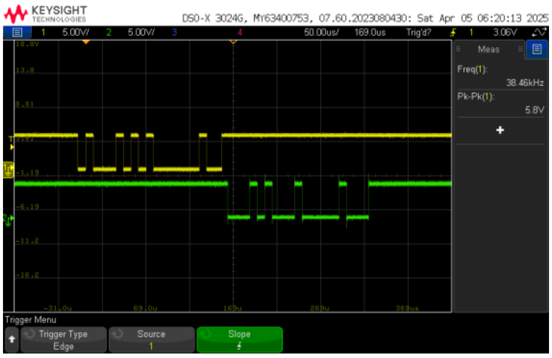


16MHz oscillator

12MHz oscillator

6. Code Flashing and UART Communication:
Code flashing through CH340G confirmed. Oscilloscope images captured on D+/D- (USB lines) and RX/TX (UART lines) show active data transfer during programming.

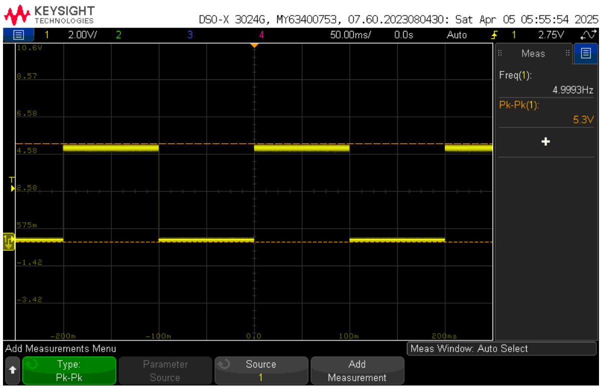




D+/D-

Tx/Rx

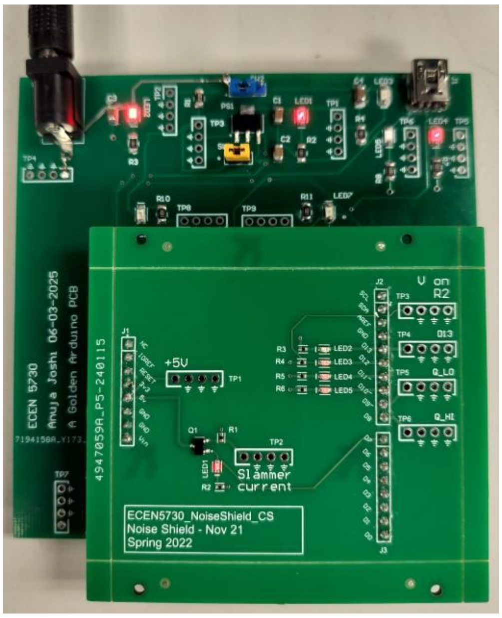
7. Flashing blinky:
Uploaded the standard "Blink" sketch via Arduino IDE. The D13 LED toggled, confirming proper MCU function and successful code execution.



Comparing your Arduino to a commercial Arduino:

We flashed the below code (shown below) onto both the commercial Arduino Uno and our custom-designed board, and then performed a side-by-side comparison.

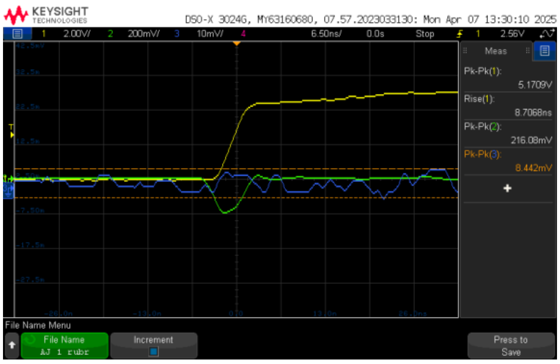
```
void setup() {   DDRB =
B00111111;   pinMode(7,
OUTPUT);
digitalWrite(7, LOW);
}
void loop() {   PORTB =
B00111101;
delayMicroseconds(4);
PORTB = B00000001;
delay(1);
   digitalWrite(7, HIGH);
delayMicroseconds(400);
digitalWrite(7, LOW);
delay(10);
}
```

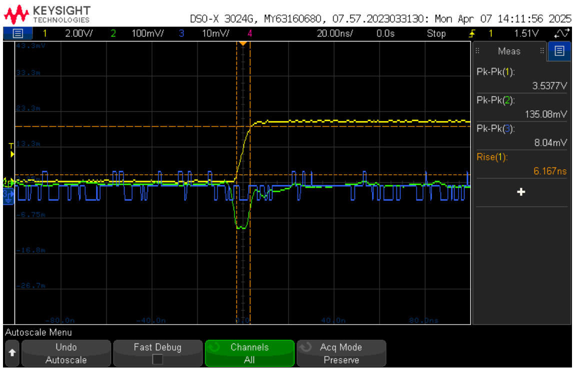


In this sketch, pins D13, D12, D11, and D10 are activated simultaneously using a direct port manipulation command. Pin D9 is configured as a quiet LOW, and pin D8 is set as a quiet HIGH. In the second phase of the sketch, pin D7 is enabled to trigger the slammer circuit. Note: on some boards, the signal controlling the MOSFET gate may be routed to a different pin (e.g., pin 1). It's essential to trace the actual connections on your board to confirm the correct control pin.

1. The switching noise and the quiet LOW pins when the AtMega switches states

Rise time measurements:





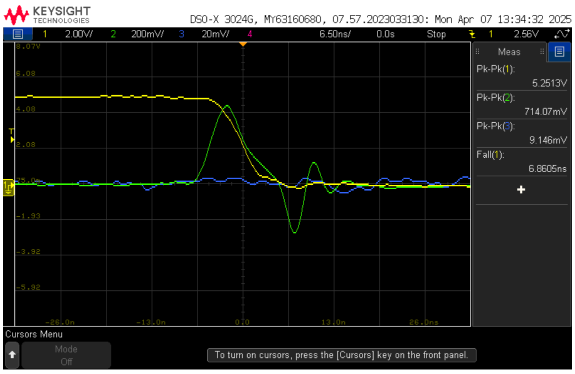
Custom designed Arduino commercial

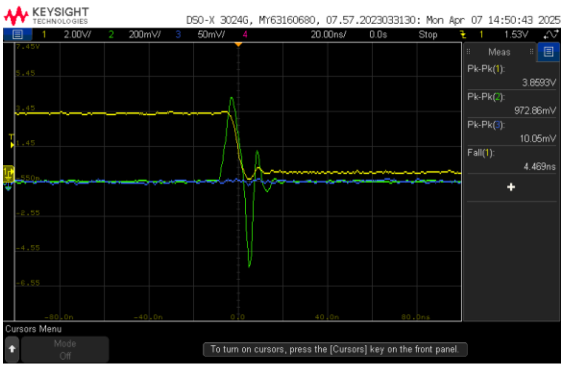
	Rise time	On die Vp-p	On board Vp-p
Custom designed Arduino	8.706ns	216.08mV	8.442mV
commercial	6.167ns	135.08mV	8.04mV

Analysis:

- The custom board shows a slightly slower rise time (8.7 ns) compared to the commercial board (6.1 ns), possibly due to longer trace lengths.
- However, the on-board Vp-p noise is nearly identical for approx. 8 mV, indicating effective layout practices in the custom design for power and ground noise control.
- Interestingly, the on-die Vp-p noise is higher in the custom board (216 mV vs. 135 mV), which may be attributed to layout-related effects near the microcontroller, such as via placement or decoupling capacitor positioning.

Fall time measurement





My arduino commercial

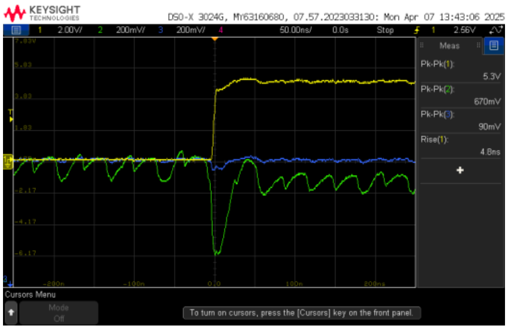
	Fall time	On die Vp-p	On board Vp-p
Custom designed Arduino	6.8605ns	714.07mV	9.146mV
commercial	4.469ns	972.66mV	10.05mV

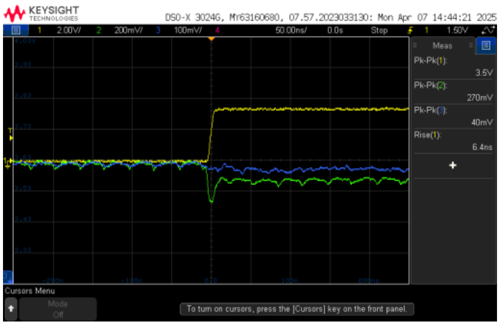
Analysis:

- The custom board’s fall time is slower than the commercial one (6.86 ns vs. 4.47 ns), which again may result from longer trace lengths or higher capacitive loading.
- However, it shows a notably lower on-die Vp-p noise (714 mV vs. 972 mV), indicating that the custom layout may be more effective in managing internal switching noise during falling transitions.
- The on-board Vp-p noise is also slightly better (9.15 mV vs. 10.05 mV), demonstrating solid PCB routing and decoupling strategies in the custom design.

2. The quiet HIGH and board level power rail noise when the AtMega switches states

Rise time:





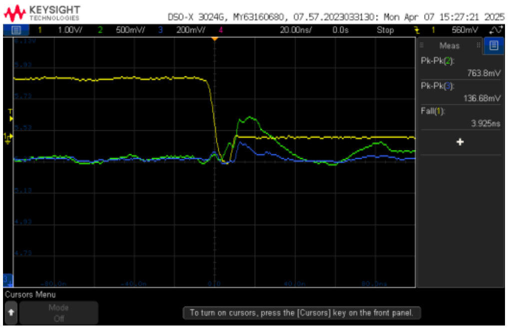
My arduino commercial

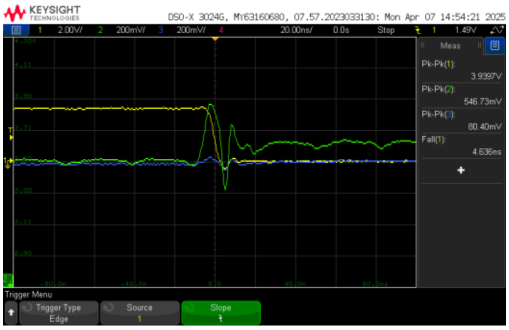
	Rise time	On die Vp-p	On board Vp-p
Custom designed Arduino	4.8ns	670mV	90mV
commercial	6.4ns	270mV	40mV

Analysis:

- The custom board shows a faster rise time (4.8 ns) compared to the commercial board (6.4 ns).
- However, the on-die Vp-p noise is higher in the custom board (670 mV vs. 270 mV), indicating increased switching noise inside the chip, possibly due to higher simultaneous switching activity or less optimal decoupling placement near the MCU.
- The on-board Vp-p noise is also higher in the custom board (90 mV vs. 40 mV), though still within acceptable range. This reflects more power rail disturbance during I/O transitions, potentially due to layout or power plane design.

Fall time





My arduino commercial

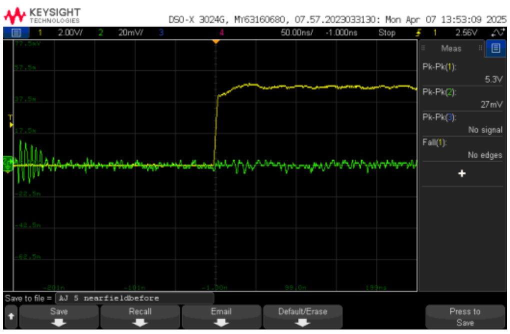
	Fall time	On die Vp-p	On board Vp-p
Custom designed Arduino	3.925ns	763.8mV	136.68mV
commercial	4.636ns	546.73mV	80.40mV

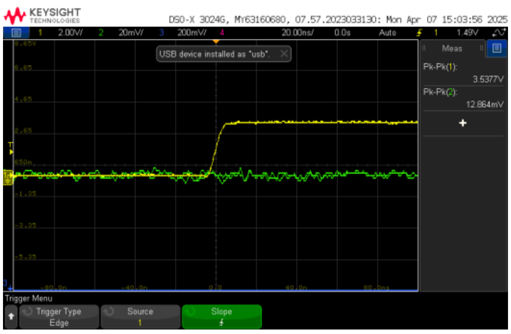
Analysis:

- The custom board again shows a faster fall time (3.93 ns) than the commercial Arduino (4.64 ns), which may reflect lower output impedance or tighter trace layout.
- However, this speed comes at a cost — on-die Vp-p noise is significantly higher on the custom board (763.8 mV vs. 546.73 mV), pointing to stronger internal switching disturbances during the fall edge.
- The on-board Vp-p noise is also noticeably higher in the custom board (136.68 mV vs. 80.40 mV), suggesting that power rail decoupling or ground return paths could be further optimized.

3. The near field emissions from under the board

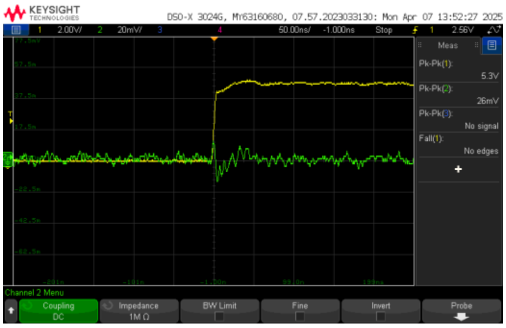
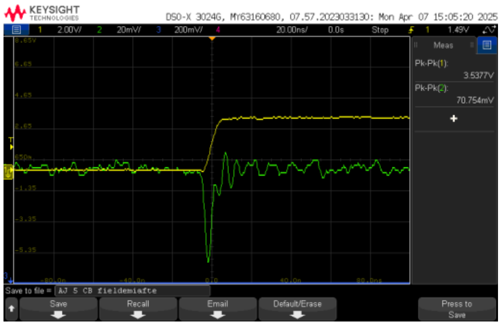
Before:





My arduino Commercial

After:

My arduino commercial

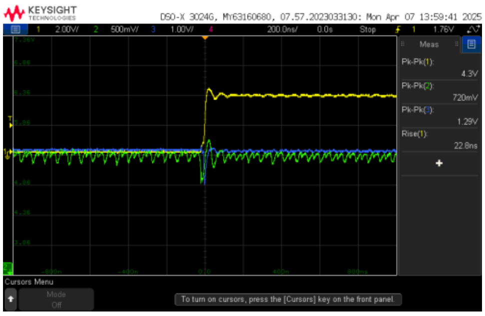
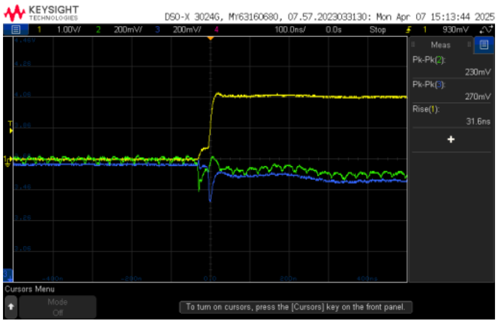
Analysis:

To evaluate emissions from the board, the oscilloscope probe was configured by shorting its tip to the ground sleeve, forming a loop for localized noise detection. During rise-time events, the probe was positioned just under the board.

It was observed that whenever the probe was placed near the board, the oscilloscope displayed a downward dip in the waveform, indicating a burst of high-frequency noise during signal transitions. This confirms that fast switching activity on the board causes localized emissions, which are detectable using this probing method.

4. The quiet HIGH and board level power rail noise when the slammer circuit switches on

Rise time:

My arduino commercial

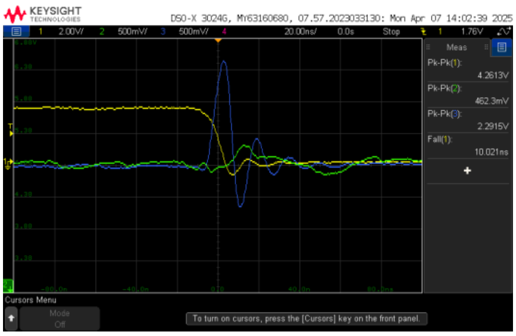
	Rise time	On die Vp-p	On board Vp-p
Custom designed Arduino	22.8ns	720mV	1.29V
commercial	31.6ns	230mV	270mV

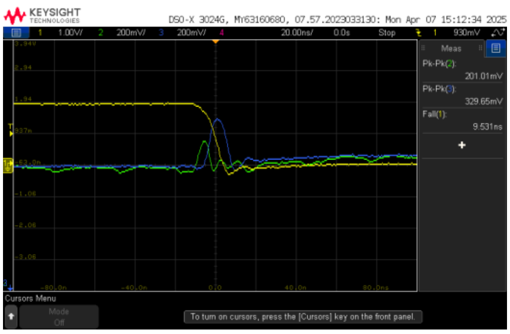
Analysis:

- The custom board exhibited significantly higher on-board Vp-p noise (1.29 V) during slammer switching, compared to the commercial board (270 mV). This indicates that more noise is being coupled into the power rail due to aggressive switching.
- Similarly, the on-die Vp-p noise for the quiet HIGH signal was also substantially higher on the custom board (720 mV vs. 230 mV), suggesting weaker internal noise isolation.

- The rise time on the custom board was faster (22.8 ns vs. 31.6 ns), reflecting stronger drive capability—but this also contributes to the higher noise, due to sharper current demands during transitions.

Fall time:





	Fall time	On die Vp-p	On board Vp-p
Custom designed Arduino	10.021ns	462.3mV	2.2915V
commercial	9.531ns	329.65mV	329.65mV

Analysis:

- The custom board shows significantly higher board-level Vp-p noise (2.29 V) during the falling edge, compared to the commercial board (329.65 mV). This large voltage swing indicates severe coupling of switching transients into the power rail and possibly poor decoupling near critical components.
- The on-die Vp-p noise is also elevated on the custom board (462.3 mV), suggesting the internal logic sees more noise from power fluctuations than on the commercial design.
- Interestingly, the fall time is slightly slower on the custom board (10.02 ns vs. 9.53 ns), possibly due to greater capacitive loading or less drive strength during falling transitions.

<u>What did not work:</u>

1. Power Jack Soldering Issue:
Initially, the power jack footprint was incorrectly designed — the 5V output pin was implemented as a simple trace rather than a proper plated through-hole pad. As a result, the power jack could not be securely soldered to the PCB. To resolve this, the 5V pin was carefully bent and manually soldered to the exposed trace. After this modification, the board began receiving power correctly. In future designs, proper footprint selection with defined pads for all pins will be ensured to avoid mechanical instability and poor soldering connections.

2. Excessive Cross Unders in Layout:
The PCB design included multiple cross unders (vias under components or signal traces routed on inner layers), which are not ideal due to added inductance and possible signal integrity issues. In future iterations, better component placement and cleaner routing will be prioritized to minimize vias and improve performance.

3. Crystal Soldering Problem:
The board failed to boot initially due to poor soldering of the 16 MHz crystal connected to the Atmega328. After identifying and correcting the soldering issue, the board started functioning as expected.

Conclusion:

The custom-designed Arduino successfully met the core functional requirements—bootloading, sketch execution, and compatibility with standard Arduino headers—while also integrating test points, power isolation features, and current sensing for deeper analysis.

Through detailed comparisons with a commercial Arduino Uno, the custom board demonstrated faster signal transitions but at the cost of higher on-die and on-board noise levels, especially under slammer circuit conditions. These results highlighted the trade-offs between speed and signal integrity in digital system design.

Key learnings:

1. Avoid Excessive Cross-Unders:
Using too many cross-unders (vias or traces routed underneath components) can increase layout complexity and contribute to higher switching noise and signal integrity issues.

2. Importance of GPIO Labeling:
Not labeling GPIOs on the PCB made debugging more difficult and time-consuming. Clear silkscreen labels greatly improve efficiency during testing and development.

3. Verifying Probe Connections:
Ensuring oscilloscope probes and ground clips are properly connected is essential for accurate signal capture and avoiding false readings.

4. Soldering and Continuity Checks:
Proper soldering of all components is critical. Verifying continuity after each step helps catch issues early and ensures stable electrical connections.

5. Power Circuit Verification:
After assembling the power supply section, it’s important to test whether the board powers up before proceeding with the rest of the assembly. This prevents unnecessary debugging later.