ECEN 3730-
PCB Design,
Connor Sorrell

Lab 9 Report: Measure cross talk between signal-return loops between signal-return loops in a special test board

Introduction:

- In this lab, we compare two different interconnections and observe how they affect the amount of crosstalk (interference) between a signal-return pair and a victim signal
- "Cross talk between an aggressor and victim signal-return pair"
    - Cross talk - the unwanted coupling of signals between adjacent wires/traces.
    - Aggressor signal = the main signal that is switching, introducing noise
    - Victim signal = a separate, nearby signal that may pick up unwanted noise from the aggressor
    - signal-return path pair = the signal wire and its corresponding ground (return) path

What's really happening?

- In this lab, we'll generate multiple fast switching signals and observe how much noise they induce onto a nearby victim signal.
- We will compare three different wiring approaches and measure their differences in crosstalk.
- **Goal: See how interconnect layouts affect signal integrity.**

BOM:

- two 10x probes w/ spring ground tips
- cross talk board supplied from TA's
- Arduino
- oscilloscope

Step 1:

We will generate simultaneous switching outputs from pins 8-13 of the arduino.
We will create three different patterns of switching outputs, using arduino code, in order to comprehensively test the cross-talk of the board. (Each PORTB command takes one clock cycle. At 16MHz, thats 62.5ns per instruction)

Pattern 1 - All pins are switching simultaneously

```
PORTB = B00111111; // Turn ON pins (13, 12, 11, 10, 9, 8) (for roughly ~200ns)
PORTB = B00111111;
PORTB = B00111111;
PORTB = B00111111;
PORTB = B00000000; // Turn OFF all pins
```

Pattern 2 - Increasing # of pins switching

```
PORTB = B00000001; // Turn ON pin 8
PORTB = B00000011; // Turn ON pin 9
PORTB = B00000111; // Turn ON pin 10
PORTB = B00001111; // Turn ON pin 11
PORTB = B00011111; // Turn ON pin 12
PORTB = B00111111; // Turn ON pin 3
PORTB = B00000000; // Turn OFF all
```

Pattern 3 - One pin switching at a time

```
PORTB = B00000001; // Turn on only pin 8
PORTB = B00000000;
PORTB = B00000010; // Turn on only pin 9
PORTB = B00000000;
PORTB = B00000100; // Turn on only pin 10
PORTB = B00000000;
PORTB = B00001000; // Turn on only pin 11
PORTB = B00000000;
PORTB = B00010000; // Turn on only pin 12
PORTB = B00000000;
PORTB = B00100000; // Turn on only pin 13
PORTB = B00000000;
```

Step 2:

Now, we will measure the voltage noise on the victim trace with a continuous return plane on the bottom layer (bottom ⅓ of the test board)

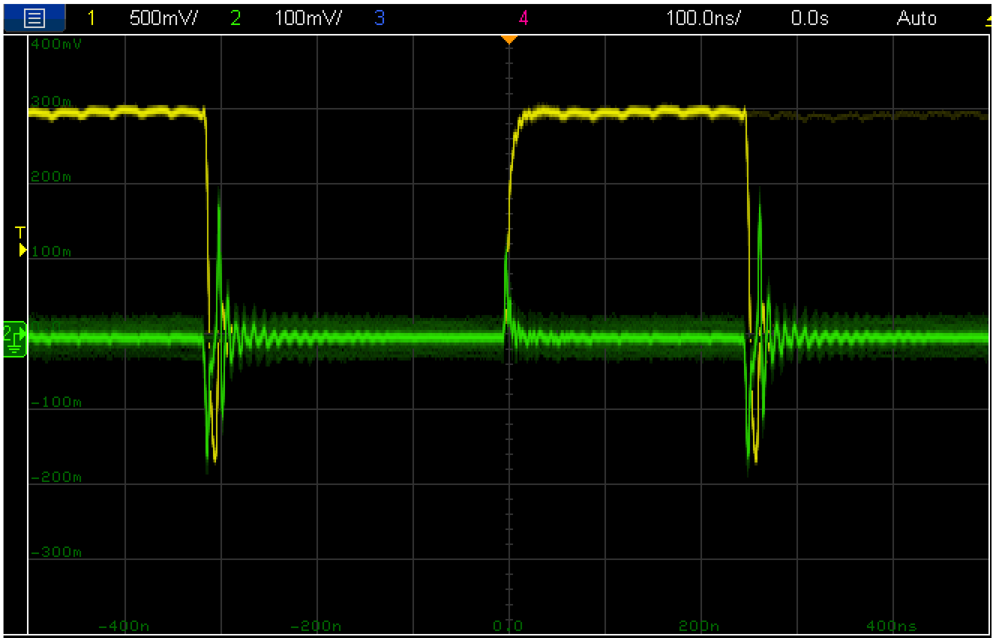

Figure 1: Shows the voltage across the 47 ohm resistor on pin 13 (yellow waveform) and the noise on the victim trace from pattern 1–when all I/O are switching on/off simultaneously (green waveform)

Figure 2: Shows the voltage across the 47 ohm resistor on pin 13 (yellow waveform) and the noise on the victim trace from pattern 2– when I/O are switching ON in increasing order (green waveform)

Figure 3: Shows the voltage across the 47 ohm resistor on pin 13 (yellow waveform) and the noise on the victim trace from pattern 3– when I/O are switching on/off one pin at a time (green waveform)

As expected, we see the most noise when all I/O are switching simultaneously.

- This is creating the largest surge in current, causing the most cross talk, due to mutual inductance
- When switching, we see a lot of ground bounce, due to the amount of return current in the ground plane.
- As all I/Os try to switch at once, the Vcc rail compresses, causing even more noise.

When we switch each I/O on in increasing order, we don't see a ton of cross talk at the rising edge, but we see that as more pins switch, the cross talk gradually increases.

- This pattern highlights how the noise increases with an increasing number of aggressors

When only one pin is switching at a time, we see the least amount of noise.

- Each edge is affecting the victim trace less because there is no combined impact of signals
- This pattern, by far, produces the least amount of crosstalk noise

From this brief experiment, we can conclude that in a real-world design, excessive simultaneous switching should be avoided. Noise increases with the number of aggressors switching together.

Step 3:

Using the middle section of the board, we will now measure cross talk with no return plane, but instead an adjacent return trace. In the following experiments, I used the arduino code that makes all pins switch on/off simultaneously.

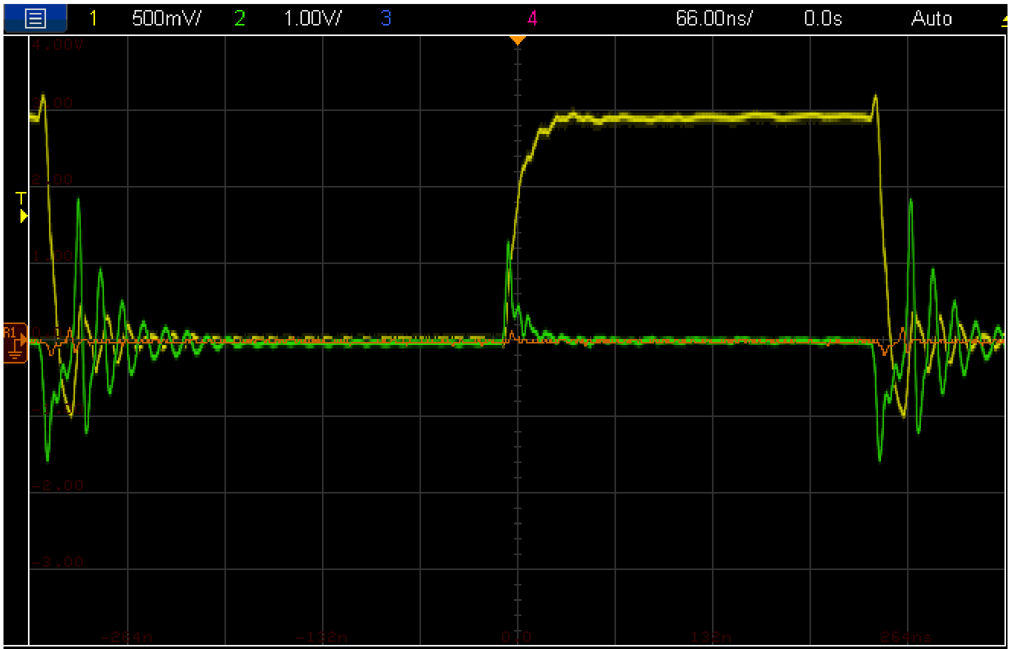

Figure 4: Shows the red waveform (noise on victim trace with a return plane) vs. the green waveform (noise on victim trace with aggressors and victim sharing a signal-return)

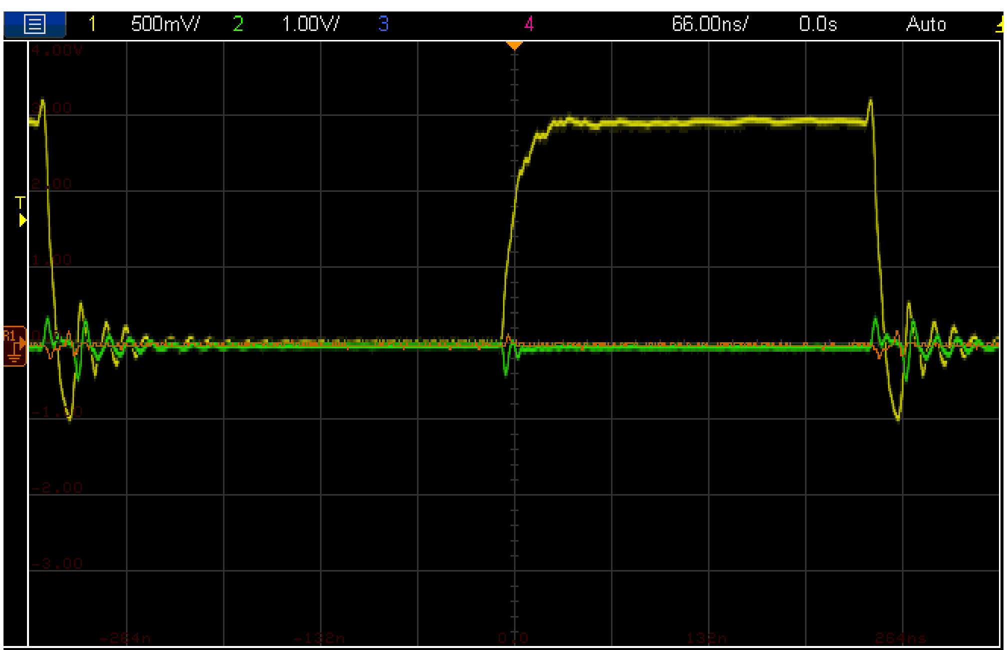

Figure 5: Shows the red waveform (noise on victim trace with a return plane) vs. the green waveform (noise on victim trace with return of victim being separate from aggressor return)

These screenshots showcase the drastic difference in noise, with and without a return plane, and with and without a shared return trace.

A ground plane:

- Acts as the return path for both the aggressor and victim signals, but follows the lowest inductance path beneath the signal trace
- With a ground plane, we see the lowest amount of noise on the victim trace
  - Noise is still present, but mainly due to power rail compression rather than mutual inductance
  - This is the preferred approach!!

A shared return trace:

- Return current has to flow through the same conductor for both signals
  - Noise coupling is much worse due to the increase in inductance
- Worst case crosstalk!!

Separate return traces:

- Victim and aggressor have their own return conductor, but no continuous ground plane
  - Return currents don't interfere as much as they do in the shared return trace
  - However, the return path inductance is still much higher than with a ground plane

<u>Conclusion:</u>

- The return path is just as important as the signal(s) themself in determining the noise
  - Ground planes are the best for minimizing crosstalk noise
  - Shared returns are the worst case possible
- Return current wants to follow the lowest impedance path

<u>Takeaways</u>:

- Crosstalk occurs when a signal wire unintentionally induces noise onto a nearby wire
- Different interconnections can reduce/greaten crosstalk
- The closer a signal is to the victim trace, the more noise we will see
- Best practices for reducing unwanted signal coupling
  - Why does this matter?
    - Poor interconnect design (which causes interference) leads to unwanted noise–corruption, errors
    - Good interconnection design minimizes crosstalk, improving signal integrity
    - This lab helps us understand how to route signals properly in a real world board design
      - Separate return paths reduce noise, but still aren’t as good as a solid ground plane which provides the lowest impedance path

- When a ground plane is not possible, use a tightly coupled signal-return path to achieve the lowest inductance.
- Keep high-speed traces short & away from each other

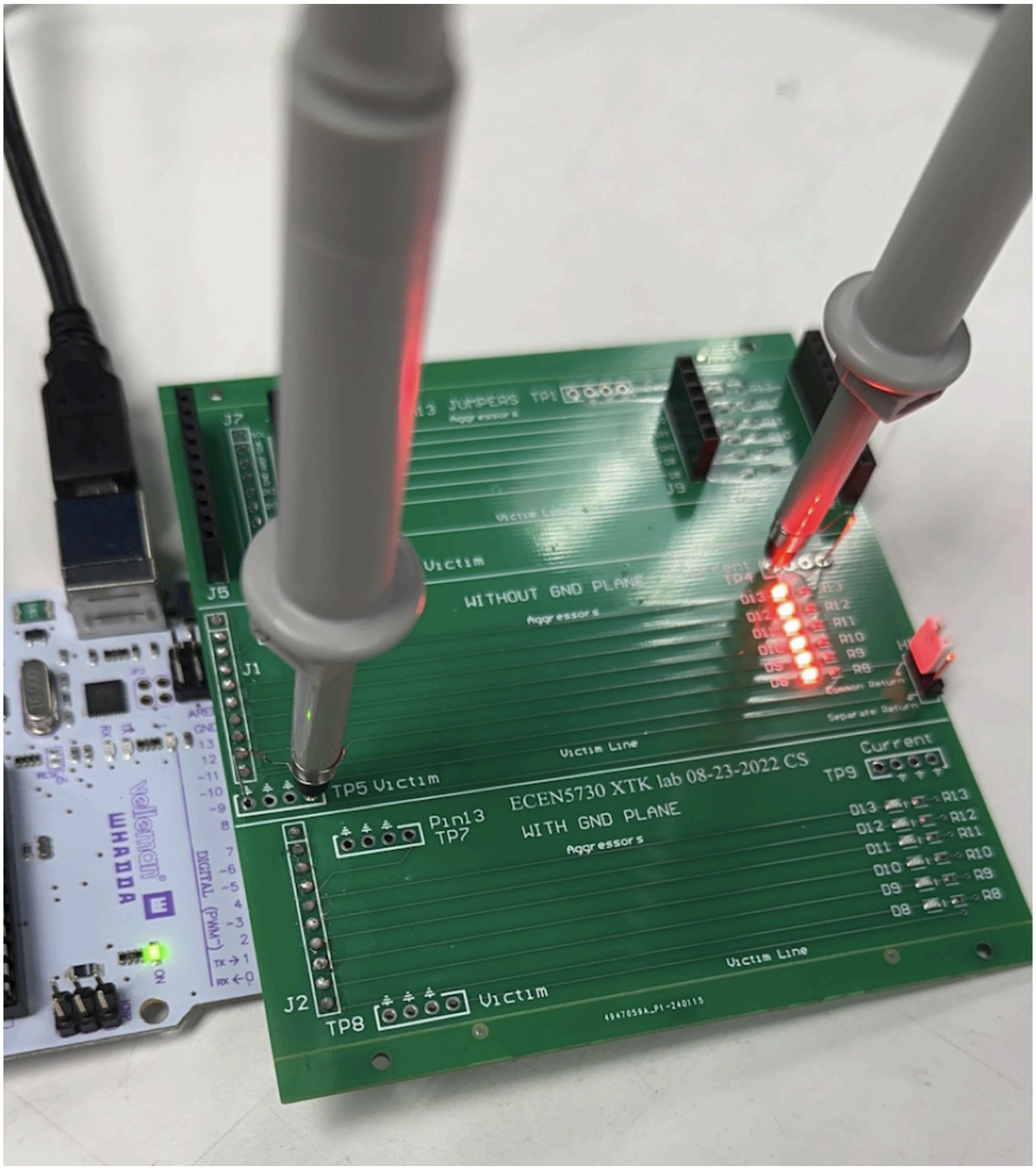

Figure 7: Shows the board layout.