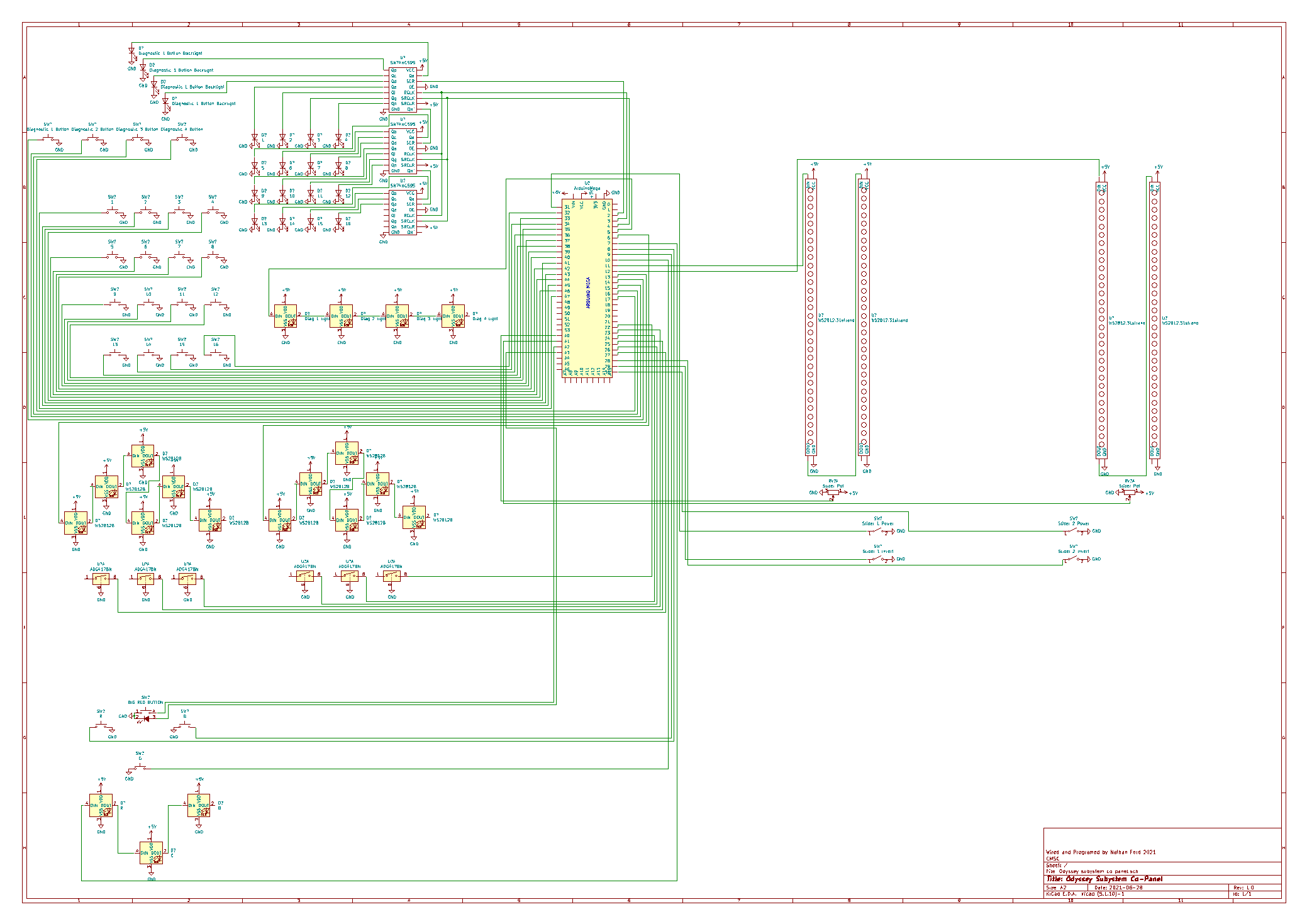
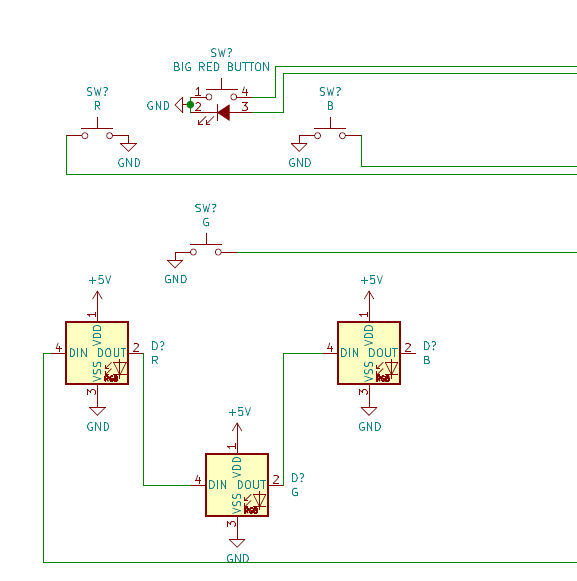
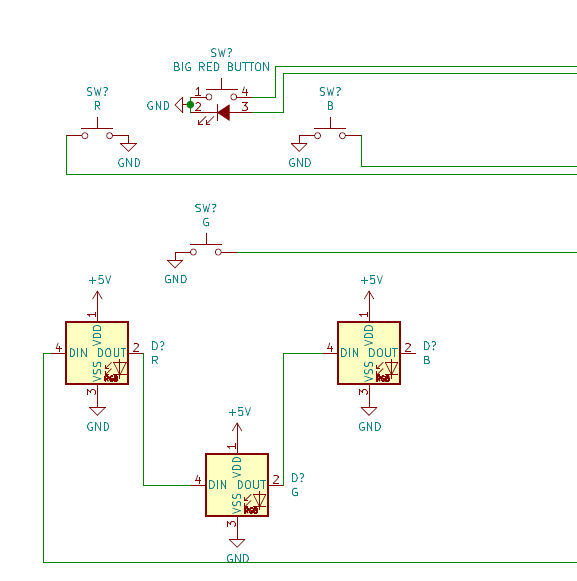
Wiring of the Odyssey Panel

****

**Table of contents:**

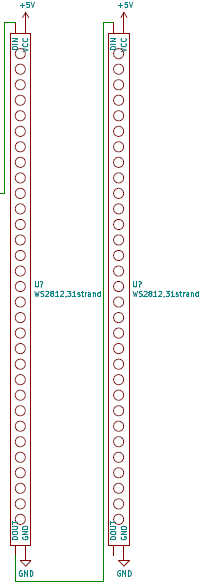
* **RGB middle buttons**
* **RGB Sliders**
* **Diagnostic Buttons**
* **Pyramid RGB lights**
* **Light-up Middle Lights**RGB Middle Buttons



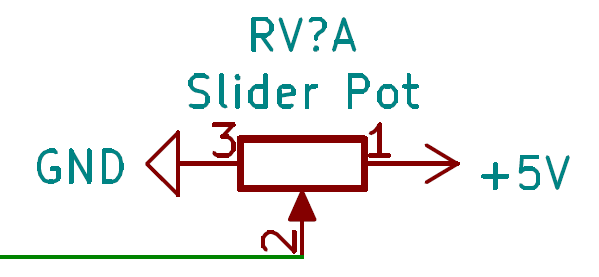
We start off with the very middle of the panel, the 3 middle triangle arcade buttons, with everyone’s favorite button the BIG RED BUTTON. The Big red button simply turns on or off a light in the middle of i… kind of anti-climactic, I know. The Big Red button is grounded, both on the negative side of the LED and on the COM port on the switch, the other side of the LED goes to A3 on the Arduino, and the Normally off pin on the switch to A2. Each of the buttons COM terminals are also grounded, and their Normally offs go to pins 8,9, and 10 on the Arduino.

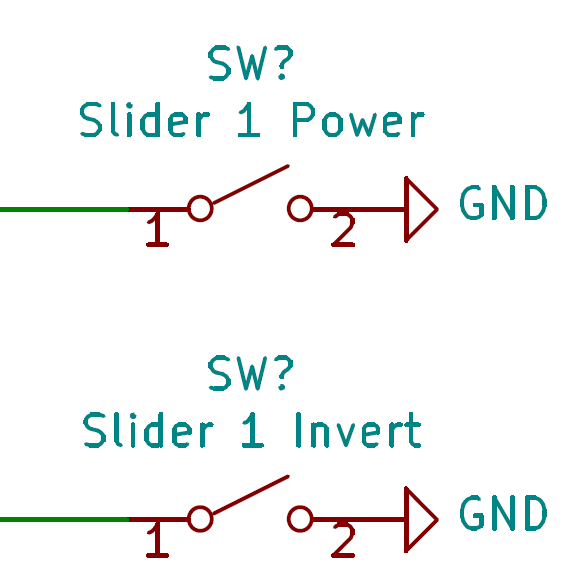
Now, for the lights on the triangle buttons. They are attached inside the buttons identically to how the buttons in subsystem are. They are wired in order of R button, G button, and B button, and are wired to pin 7 on the Arduino.

RGB Sliders

The amazing RGB sliders, these are quite fun to move around, this part consists of 3 parts, the sliders, the Strands of lights, and the switches. This part of the panel is completely symmetrical, so I will explain one side, and then pins of the second, and you will just know to mirror it.

We start off with the strands, each side of the panel has 2, they will work in sync. These will display what position the slider is in and indicate if it is turned on or off, giving the user feedback. These are hooked to pins 11 and 12 on the Arduino.

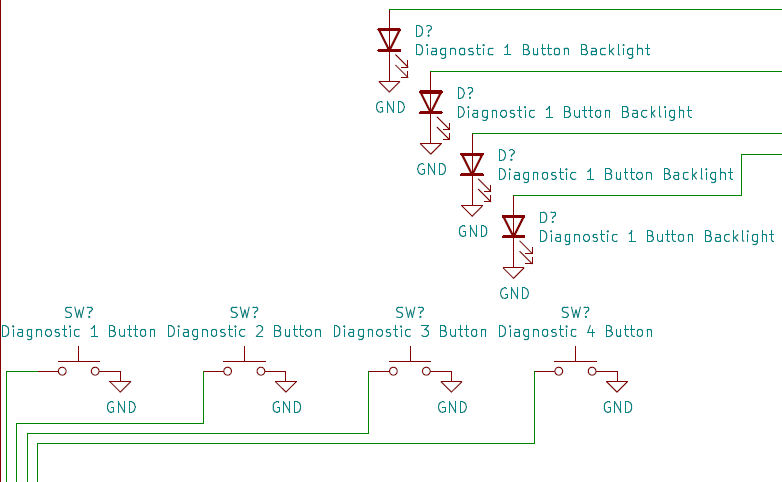
Next off we have the sliders, they are potentiometers, and so their schematic symbol is a bit underwhelming, but they are what the user will use to change the strands with. Their connections are that the com ports are hooked to pins A0 and A1 on the Arduino, and their other sides are connected to 5V and GND.

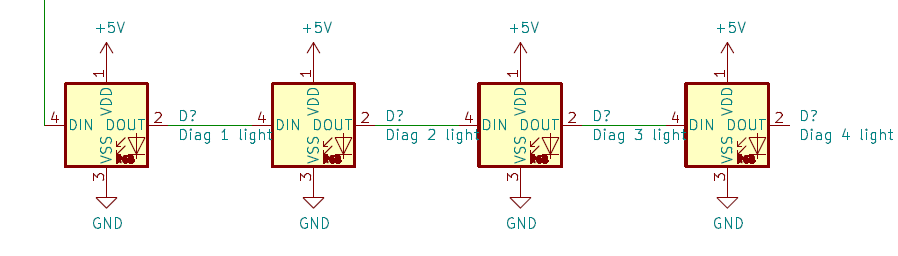


Lastly, we have the switches, these are what shakes things up, we have the power switch, which turns on or off the lights on the strips, next we have the Inverse switch, this turns the bars shown on the strands yellow, instead of blue, and has a blue segment jump up and down inside of them. Each switch is connected to ground on one side, and then for the power switches, hooked to pins 31 and 30, And for the inverse switches to pins 29 and 28.

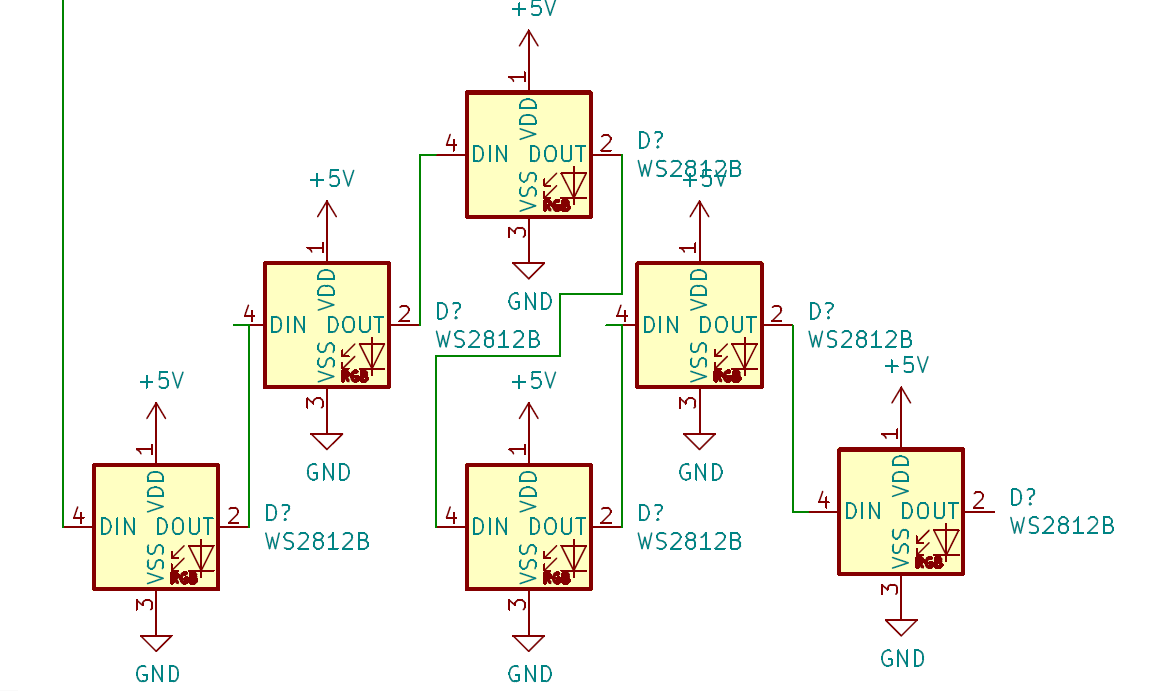
Diagnostic Buttons

The ship is malfunctioning, but we do not know why, Run a diagnostic! The diagnostic portion of the panel consists of 4 light up buttons, and 4 WS2812 lights (Neo Pixels).

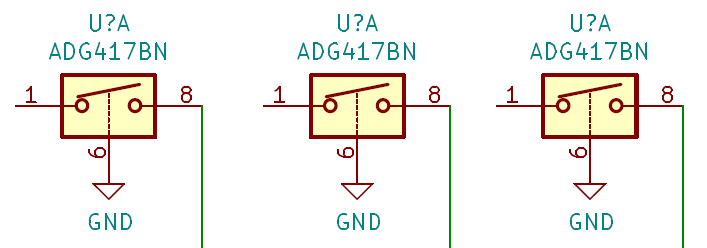
For the 4 buttons we have non-latching pushbuttons with backlghts. The buttons backlights are hooked to ground and the first 4 outputs of the shift registers, if I could go back in time, or if you are experiencing issues, please use NPN transistors between the LED’s and the shift registers and hook the LEDS to power and ground through the transistor (Like the Galileo panel), the buttons are hooked to ground and to pins 14-17 on the arduino (diag 1-4 respectivly)

Now for the neopixels, these are the feedback to the diagnostic portion, these are daisy chained from pin 5 on the arduino, to the diagnostic 1 button, all the way to the diagonsic 4 button. To make these look nicer on the panel, we attached them to the back of a classic 3.5mm led that was clear, to make it look three dimensional, one problem with this is that kids kept on pushing it in, but after superglued they don’t move an inch, and are super secure.

Pyramid RGB Lights

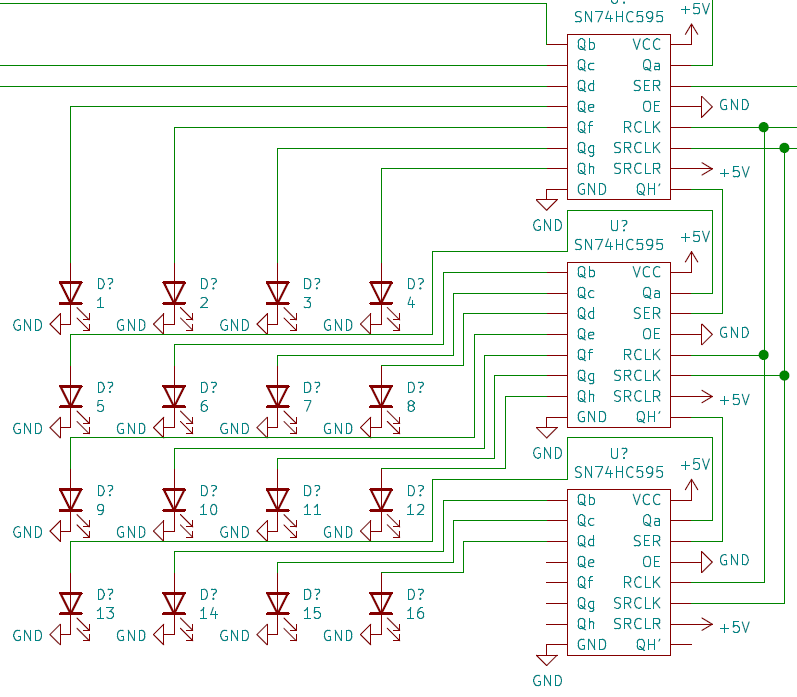
 This is another Symmetrical portion of the panel, so I will just give you pins for both sides again, just remember there is two sides. These are a fun animation, depending on how many switches are flicked change how the pattern is shown and what colors and how many colors are bouncing around in the triangle.

For the neo pixels, they are daisy chained like is shown here, and you just must give them all power and ground, they are hooked to pins 13 and 6 for each side on the Arduino.

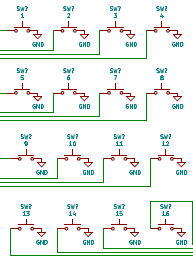
For this portion of the panel, you will need 3 switches for each side of the panel, these are hooked to ground on one side and pins 22, 24, and 26 (23, 25, and 27 for second side) on the arduino.

Light-up Middle Buttons

You see that you get a combination of red, green, blue, red from your diagnostic, you reach for your manual to see how to fix it, and it tells you to draw a smiley face on the grid to fix it… wait, maybe not quite like that for a fix, these buttons are in the middle of the bottom portion of the panel when you press them they toggle if their built in light turns on or off.



If you could not guess, we don’t have enough pins to wire ALL of the buttons lights individually, so we opted to use the shift register to control them, they are connected on the 5th input to the 20th input coming from the shift registers. I stand by what I said above about the diagnostic Backlights, and if you are having issues, use NPN transistors between the shift register like on the Galileo.



The buttons are slightly easier to wire, less of a headache, they are hooked one side to ground and the other side to pins 32-47, but not quite in that order.

Row one: 47,45,43,41

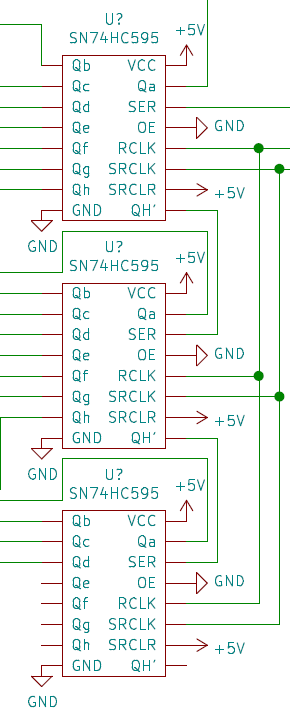
Row two: 46,44,42,40

Row three: 39,37,35,33

Row four: 38,36,34,32

This is so that each row is a 1x4 ribbon cable going to the arduino, I found two wide cables are a pain to put in, so this makes it easier.

Shift Registers

 Now we need to know how these magical shift registers are hooked to the Arduino, now I want you to keep in mind with the 3 registers we have set up this can output up to 24 different outputs, and it only requires 3 inputs, crazy, right? These connections are SCK (Serial Clock), MOSI (Master Out Slave In), and finally CS (Chip Select). The CLK pins are connected all between the shift registers to pin 4 on the Arduino. Similarly, the CS pins of all three chips are hooked to pin 3 on the Arduino. The data gets a little more complicated, you have to send the data from pin 2 on the Arduino to the MOSI/SER pin on the first register, then we take the signal from QH’ and hook that to the MOSI/SER on the second chip, and we continue that down the line. We then must apply 5V power to VCC (power) and SRCLR (clear). Then we hook GND to the GND pins and OE (Output Enable) for it to work. Qa-Qh are the 8 outputs for each chip. Also, if it helps these schematics line up exactly with the position of the pins on the real chip.

