ETUDE II: PERCEPTRON-P

CART 360, AUTUMN 2020
Martin Hanses

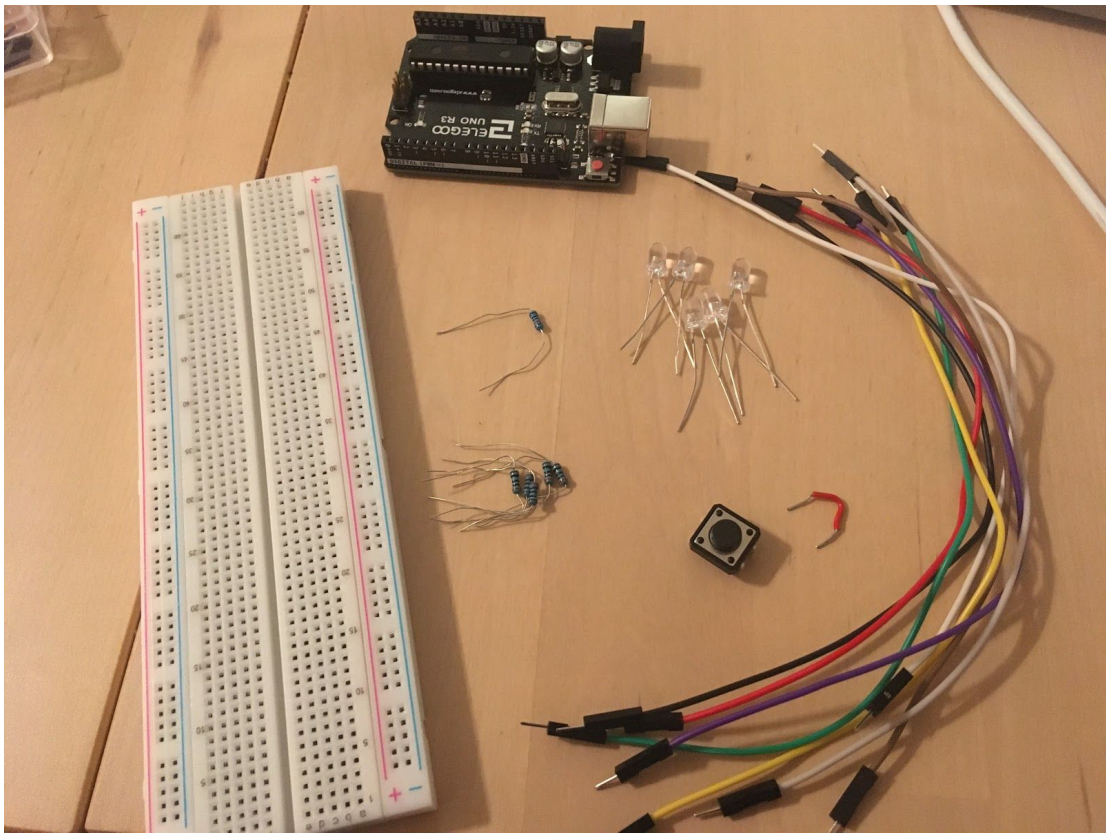
1. BUILDING

I started off by studying the picture and getting out all the parts needed for the build. I've gotten used to tracing the circuit by eye, and based on the first look, it seemed like a few of the parts were unnecessary - some weren't used and some were just excessive.

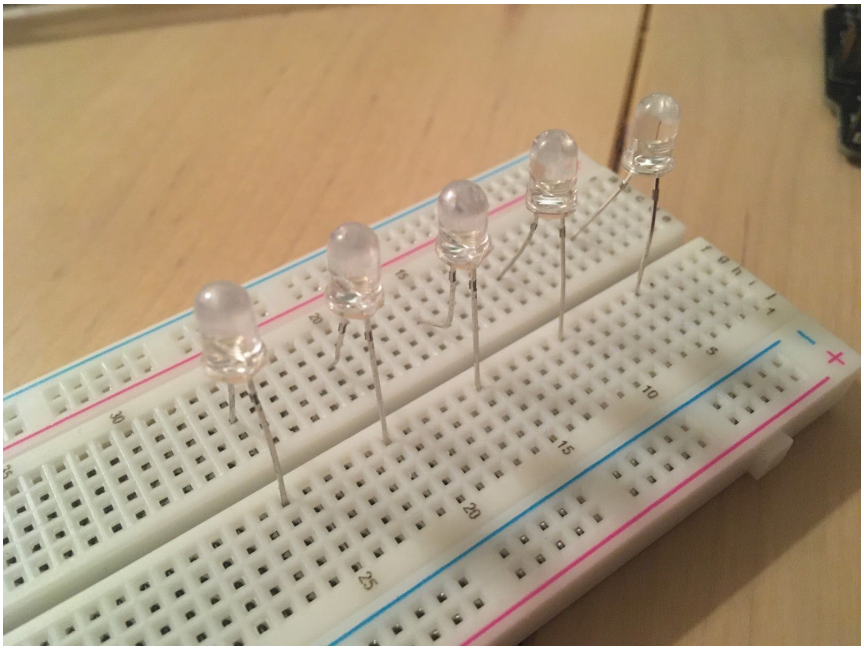
The first thing that seemed off to me was the use of connecting wires between the two halves of the breadboard - it seemed reasonable to me to just connect the LEDs on either side and just not deal with five additional tiny wires. Additionally, the connecting ground/power wires between the two halves seemed unnecessary, so I wanted to try without them first.

The following are what I thought I'd need:

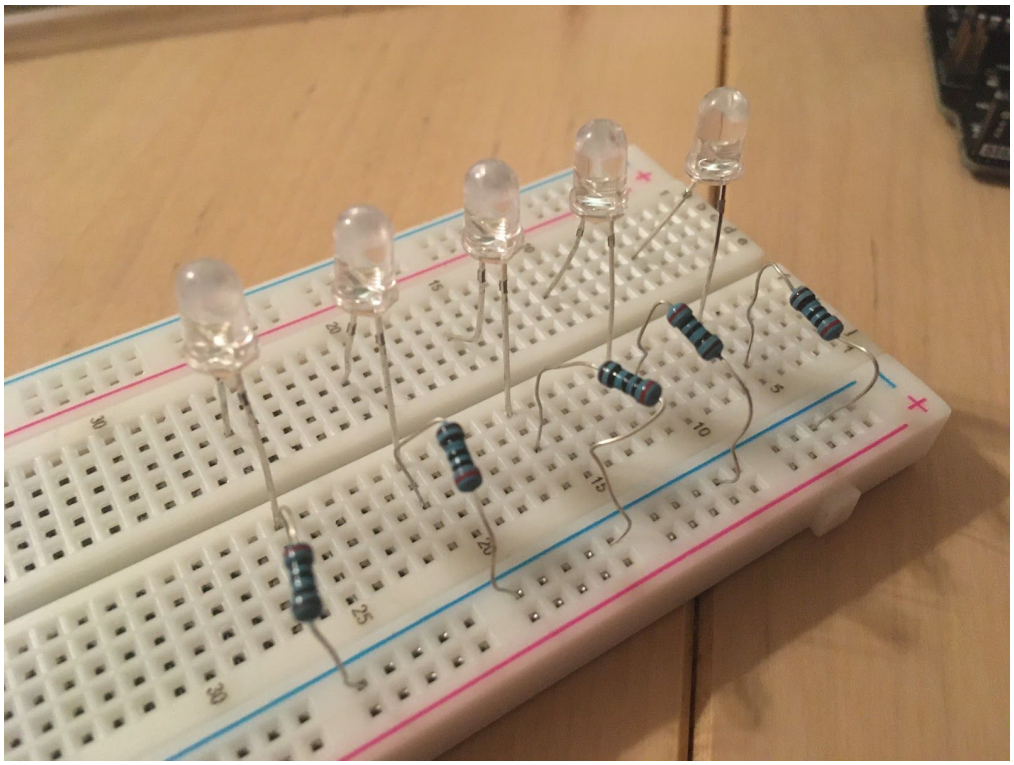
5 Clear LEDs
8 cables
2 medium-length wires
1 short-length wire
1 button
1 10k OHM resistor
5 200 OHM resistors



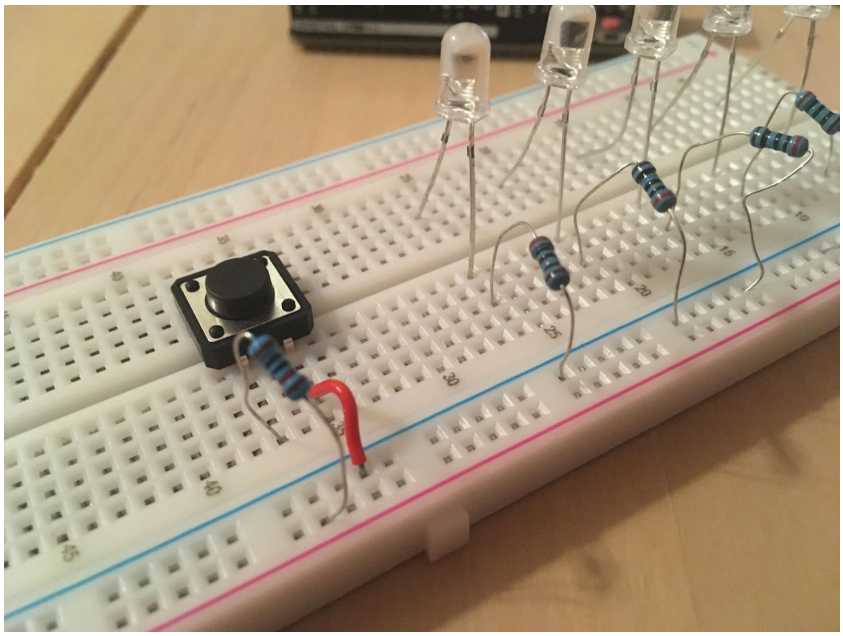
I started by making sure the LEDS fit nicely as they connected the two halves, and thought of putting them all on one side of the breadboard. I figured that would have been a little too tightly packed, so went on with the original plan.



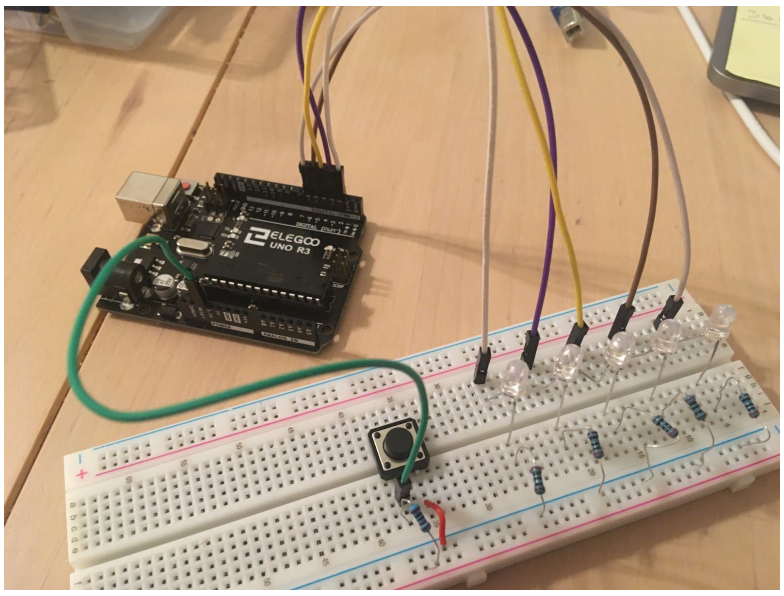
Next came the 200 OHM resistors, since they were all paired up with an LED each.



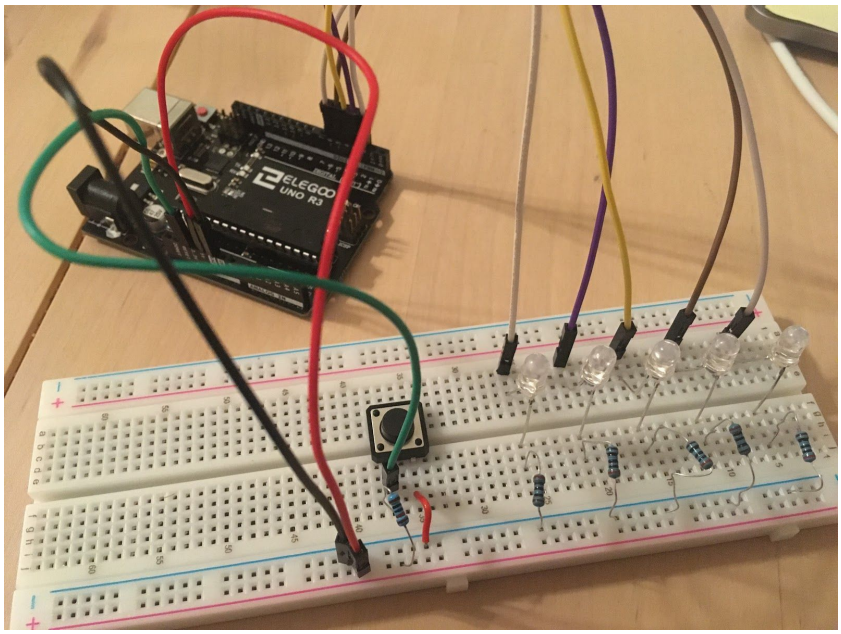
I figured I'd go on with more components that fit snugly on the breadboard, so I connected the button, the short wire, as well as the 10k OHM resistor next.
(I'm usually paranoid about electricity and electric currents, so I often leave the breadboard and Arduino without power until I've connected everything else.)



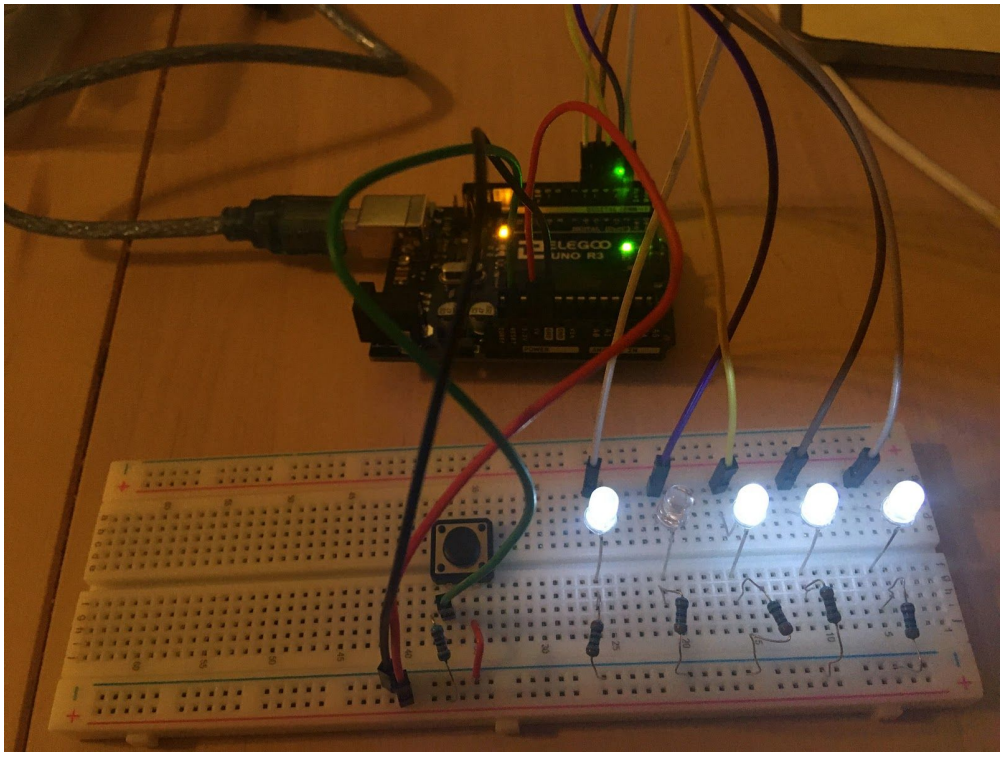
I've started to dislike the basic wires in our kit, so I opted to use the other kind of wiring for this project. (And some before it.) I connected all of those next, except the 5V and ground wires.



Finally, I added in the power wires. Once again, I traced the circuit with my finger and figured it would work fine like this.



Next, I connected the Arduino to my laptop and ran the code, and lo and behold, the LEDs started flashing. (I actually got lucky and took this picture as one of them was off, which I thought was an interesting look into how the project is intended to work.)



2. ALTERNATE CIRCUIT

The major difference between the two is the resistor/s - the first is wired with parallel resistors, while the second only uses one. Both circuits use 200OHM resistors in the areas that are different. This has a direct impact on the resistance of the circuit, since the version with a single resistor simply has a resistance of 200 OHM, and the parallel resistors total 40 OHM. (Parallel resistors have an equivalent resistance of 1/Req = 1/R1 + 1/R2...)

More resistors means more channels for the current to flow, which in turn means a higher current, which in turn means brighter LEDs. The more reliable option is the first one, since it runs in parallel and with a current appropriate for the LEDs.

The current in Version 1 (parallel resistors) equals 5/40 (V/R) = 0.125A (125mA)
The current in Version 2 (single resistor) equals 5/200 = 0.025A (25mA)

3. RED RECTANGLE

The code used in the device sets the numbered PINS to OUTPUT, which means they provide power to the LEDs in the circuit. The power connection from the red cable leading from 5V in the Arduino is only there to manage the button connected to the RESET pin.

Arduino UNO specifications tell us that when the RESET pin is set to LOW (pulled to ground), it has the same function as the red reset button on the microcontroller, triggering a reset. Using a pull-up resistor (the 10k OHM resistor pictured in the red square), it is constantly set to its high state, meaning no reset is triggered.

When the button pictured is pressed, the line goes to ground, meaning it triggers the low state of the line and thus the reset through the RESET pin.

4. NOTES ON PHOTOS AND VIDEO:

I ended up having a lot of issues taking good pictures of the effect. Standing up with the camera and moving it from side to side, while having someone else taking pictures, just didn't seem to work. The only way I got any legible text out of the pictures was by angling the perceptron in a particular way, and surprisingly, taking pictures from above in a well-lit room (my kitchen, next to where I work). This might have been a side-effect of my not-so-great camera phone, and most likely, the results would have been better with a professional camera. Unfortunately, I had to make do with the best I had. That said, the effect is clearly visible in the attached pictures - just not the whole phrase at the same time. The full altered text is "IM MARA !", as well as a bonus "OOOO !" for halloween spookiness. I decided to make the new text short and easy to track, since I was having issues taking pictures of the effect.