CART360
Etude Two

Sept. 26th 2019

Part one

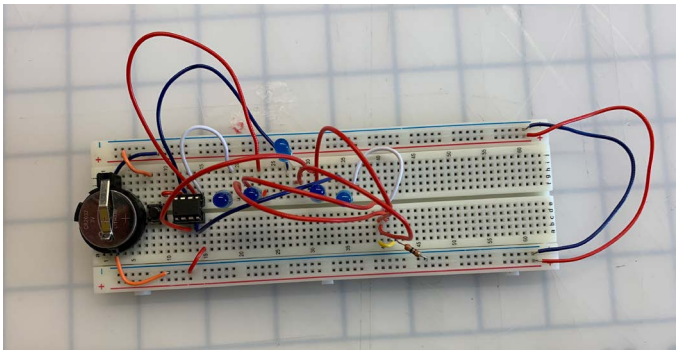
For this first part, I started to build an early version of the circuit on my breadboard. It took me time to understand how to connect everything one to another. I did the first try by myself, and once I added the battery, nothing was working. With some help, I have come to realize that I did not connect the negative part of the LEDs to the resistor. I made a new version of the circuit with everything connected at the right place. The LEDs were now lighting up correctly. I added to the code the pin, and it was now time to build the circuit on the right board.

I taught that my welding techniques were good enough to build the circuit without a problem. I sketched the circuit to make sure I understood the connection correctly. Once I was confident enough, I started building on the breadboard. In the middle of the process, I realized that I should have practiced my technique beforehand. I had a lot of difficulty to connect every piece together without making a mess on the breadboard. In the beginning, everything was going well, but once I had to solder the negative part of the LEDs with the pin, everything went wrong. The space between the resistor and the LEDs was small, and I solder everything together in the wrong way. I ended up with a mountain of material. I tried to make it a little better by taking some part of and rebuilding it, but the more I was trying to make it better the worst it was getting. After many tries, I ended up having two functioning lights. Also, I connected some elements on top of each other, which I think it not an efficient way to make my circuit work. These two problems can be the reason why I only have two working LEDs.

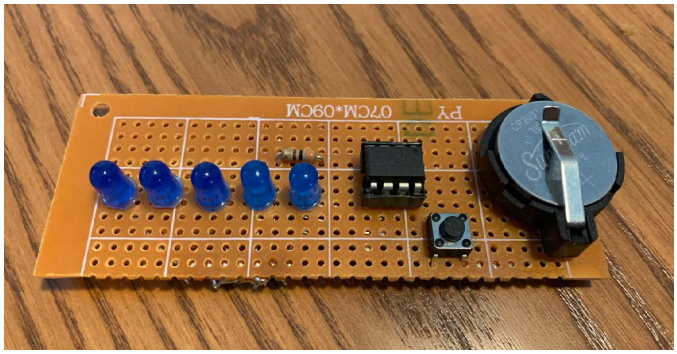
I added the battery to the circuit and only had 3 LEDs lighting up. After looking up with more attention to the breadboard, I saw that my mountain of solder was not connecting correctly to the resistor. We can see holes in the mountain, which can explain why two lights can't close the circuit. Overall, I understood the circuit and how to make every connection to make it happen. The main problem of this etude was that I did not practice my soldering skills enough before doing the final breadboard.

CART360
Etude Two

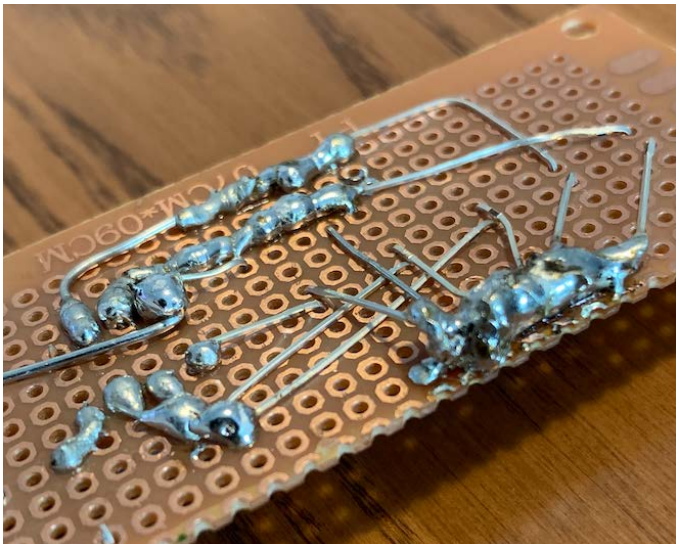
Sept. 26th 2019

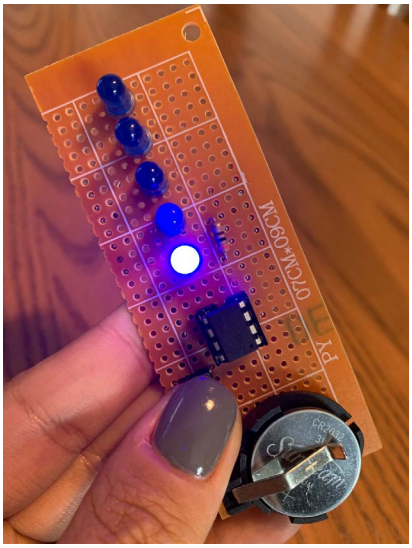


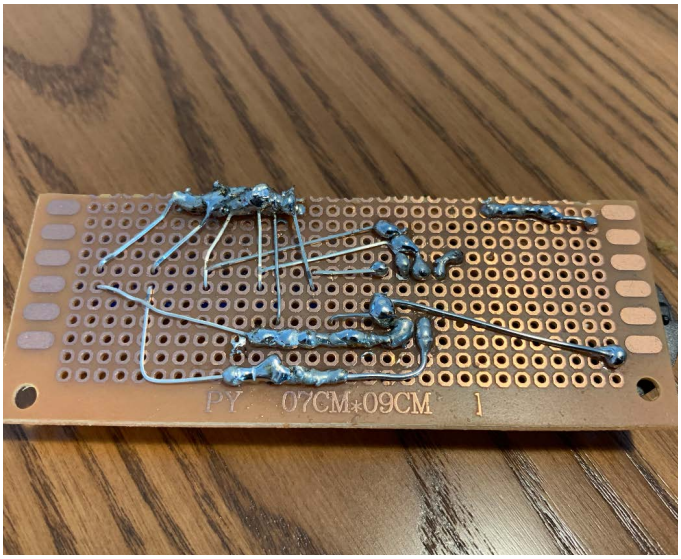
Building the circuit on my own breadboard
We had to change the place of one of the LED because my breadboard was not working correctly.

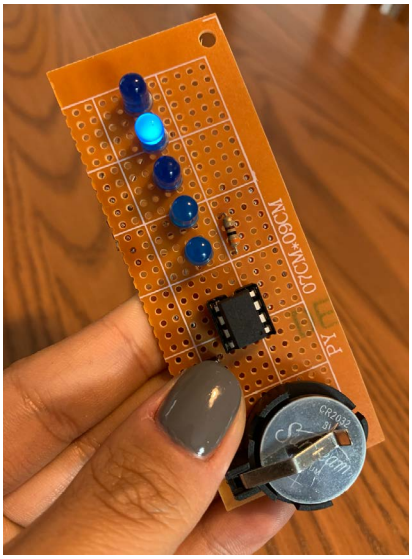


Building the circuit and soldering it
This is what the top of my circuit looks like once everything is in place.









The final product
We can see on the left that my circuit is not connected proprely. Some parts are going above others and we can also see the soldering mountain.

CART360
Etude Two

Sept. 26th 2019

Part two

I would say that the first circuit is in series and the second one is in parallel since all the LEDs have their resistor. I think the second circuit is the better one to take because if something happens to the resistor, the LEDs will continue to work. The current only have one way to go in the first one compare to the second one where the current can go around. To make the experience more meaningful, I would add another parallel circuit next to it with other LEDS that would light up after the first circuit. Like one character would be shown on the first circuit and then another one would be display right after on the second one. Maybe this way, the word would be easier to read the secret message.

Part three

For my character, I have decided to add a small smiley face to the characters. If I understood correctly, the character is display in a grid of 5x5. If we take the letter «A» for example and the code that goes with it {1, 6, 26, 6, 1}, we can understand that each square of the grid has a value. By drawing the letter «A», we can see that the value in the brackets is the result of an addition of each square of the grid that is supposed to light up. With this in mind, if my circuit would have been working, the code to had my character would have looked something like this.

Initiating the character :
int smiley[] = {10, 0, 31, 20, 8};

Displaying the character :
if (c == ":P"){for (int i = 0; i <5; i++)
{displayLine(smiley[i]);delay(delayTime);}displayLine(0);}