Yang Hu

Professor Peter Jansen

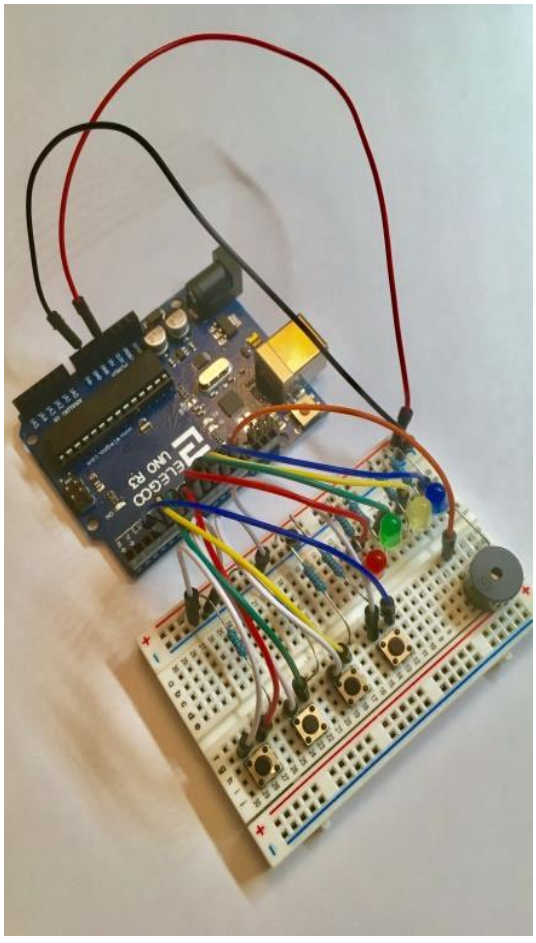
ISTA-303: Introduction to Creative Coding

26 September 2018

Simon Says Memory Game

Design Documentation

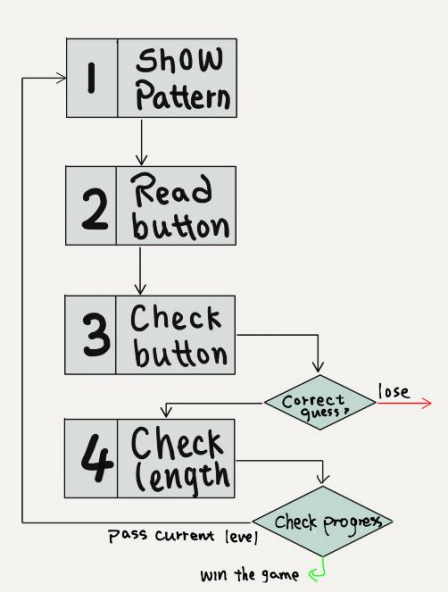
Because the version I showed in the afternoon is not completed, I will firstly describe the completed one here:



Game procedures:

1. **Game start**. All Diodes will light up together. A new answer is generated, and guesses will start from length 1.
2. **Show pattern.** This comes with diodes lights up and buzzer makes sounds (each color has a fixed pitch).
3. **User input.** Correspond light and sound will also be produced. Also, I fixed the buttons layout, they are no longer inverted.
4. **Check input.** Check after each input. Either right or wrong will have a sound feedback. If right, game proceeds to next level. If wrong, game restart.
5. **Check length.** Go to next level or game win.

Design rationales



I build my code according to the "current state" structure from the class. Basically, I break the program down into modules, and build and test them one by one.

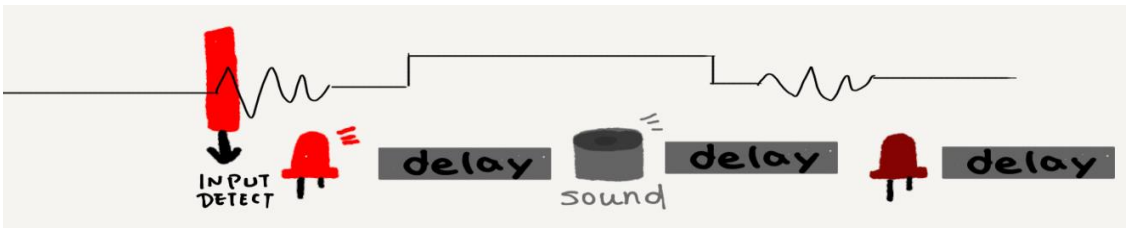
1. **Generate random answer**. The code I used:

```
// generate a random seed
// read from unconnected pin to
// get unkown result (0-1023)
randomSeed(analogRead( A0 ));
for(int i = 0; i < FINAL_LENGTH; i++){
  ANSWER[i] = random(LED_START,
                     LED_START + LED_RANGE);
}
```

Because the random function generates a fixed seed. The answers become identical during the program life cycle. Therefore, I used analog.read to read an unconnected port to get the unstable result. According to the Arduino reference,

> "If the analog input pin is not connected to anything, the value returned by analogRead() will fluctuate based on a number of factors (e.g. the values of the other analog inputs, how close your hand is to the board, etc.)."

2. **Debouncing**. To solve bouncing, I tried to deal with it by delay(). However, it assumes the button is pressed and held for a "rational" time interval. Basically, it works like this:



"AnalogRead()." *Arduino - Introduction*, www.arduino.cc/reference/en/language/functions/analog-io/analogread/.