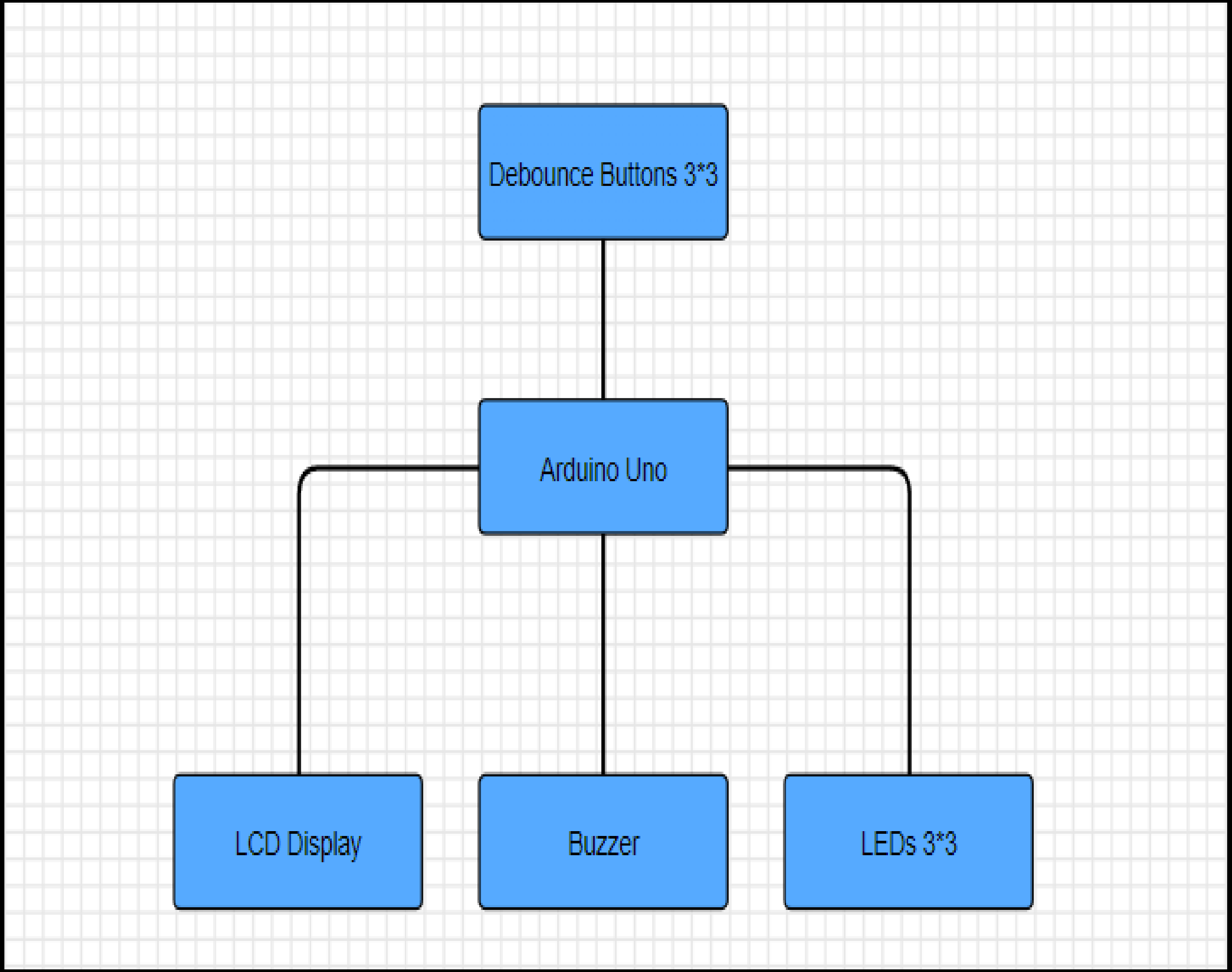
PY371 Whac-A-Mole

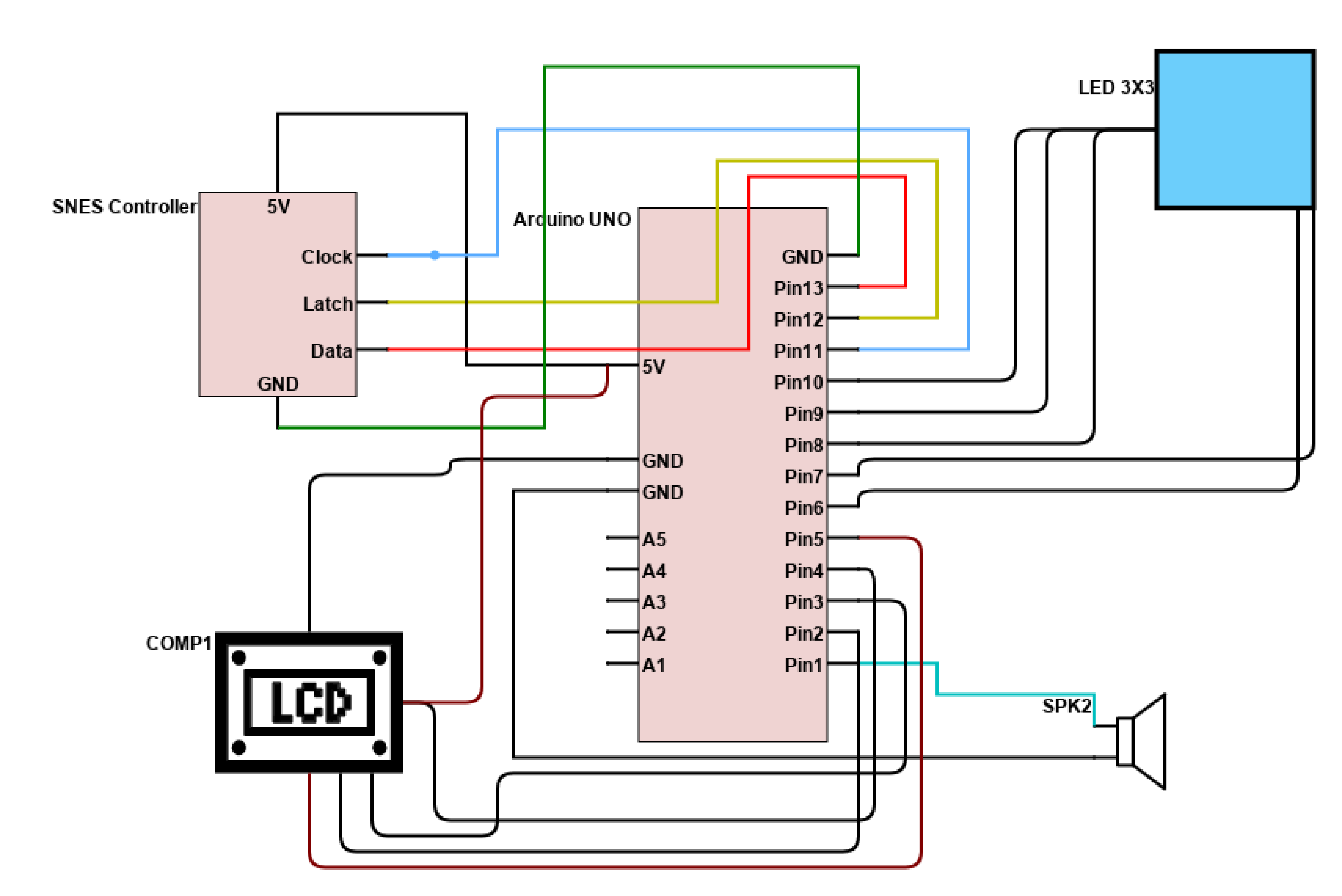
Zechen Zhang and Cai Xu

Abstract

- Whac-a-mole arcade game
- Thru SNES Controller with 8 buttons upfront
- 8 Moles represented by 8 LEDs on breadboard
- Hit as much as you can in 1 min
- Buzzer generates specific tones for scores and misses
- Scores will be calculated and demonstrated on the LCD screen
- Highest score displayed

Block Diagram

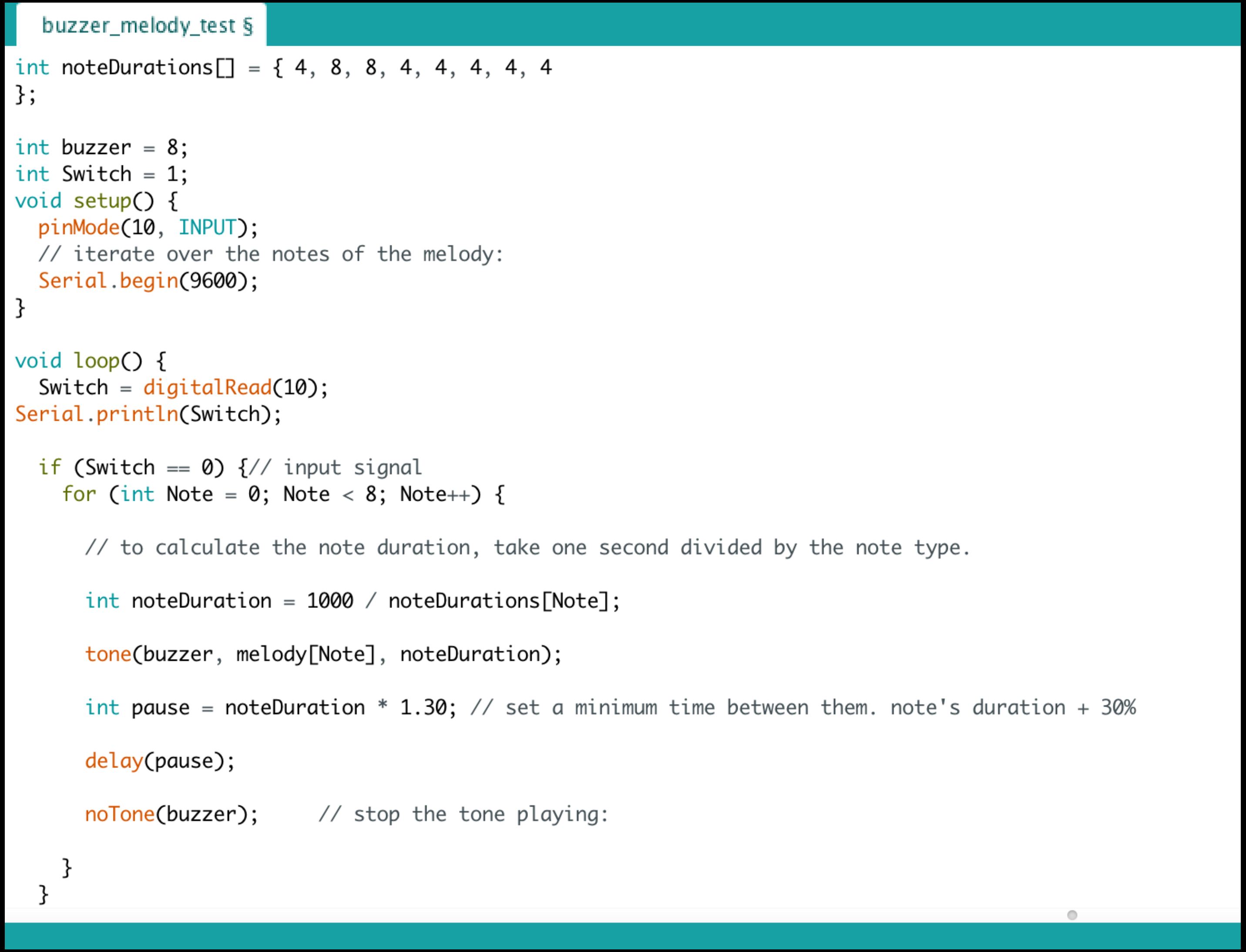



SNES Controller
5V
Clock
Latch
Data
GND
Arduino UNO
GND
Pin13
Pin12
Pin11
Pin10
Pin9
Pin8
Pin7
Pin6
Pin5
Pin4
Pin3
Pin2
Pin1
5V
GND
GND
A5
A4
A3
A2
A1
LED 3X3
COMP1
LCD
SPK2

Sound effect

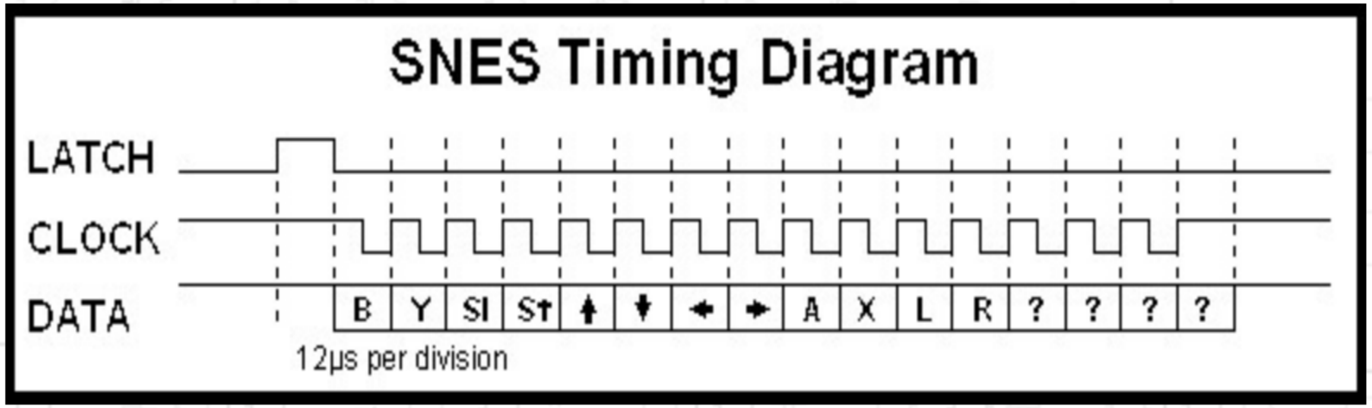
- We successfully constructed a circuit to emit sound through a 0 or 1 signal by using passive buzzer.

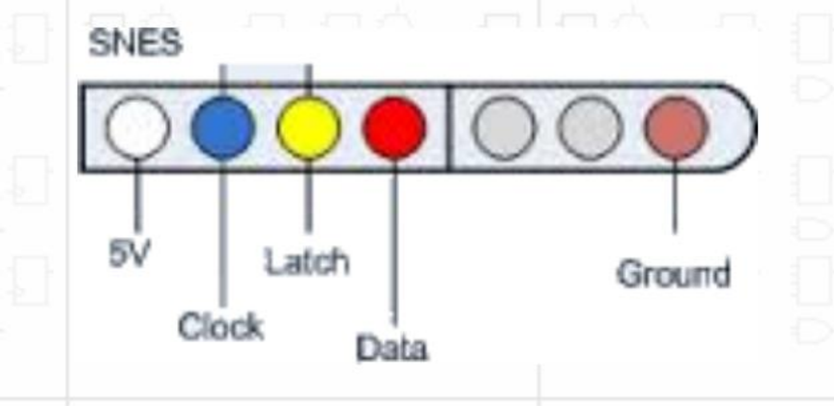
- Timeline : Building a random matrix to processing signals.

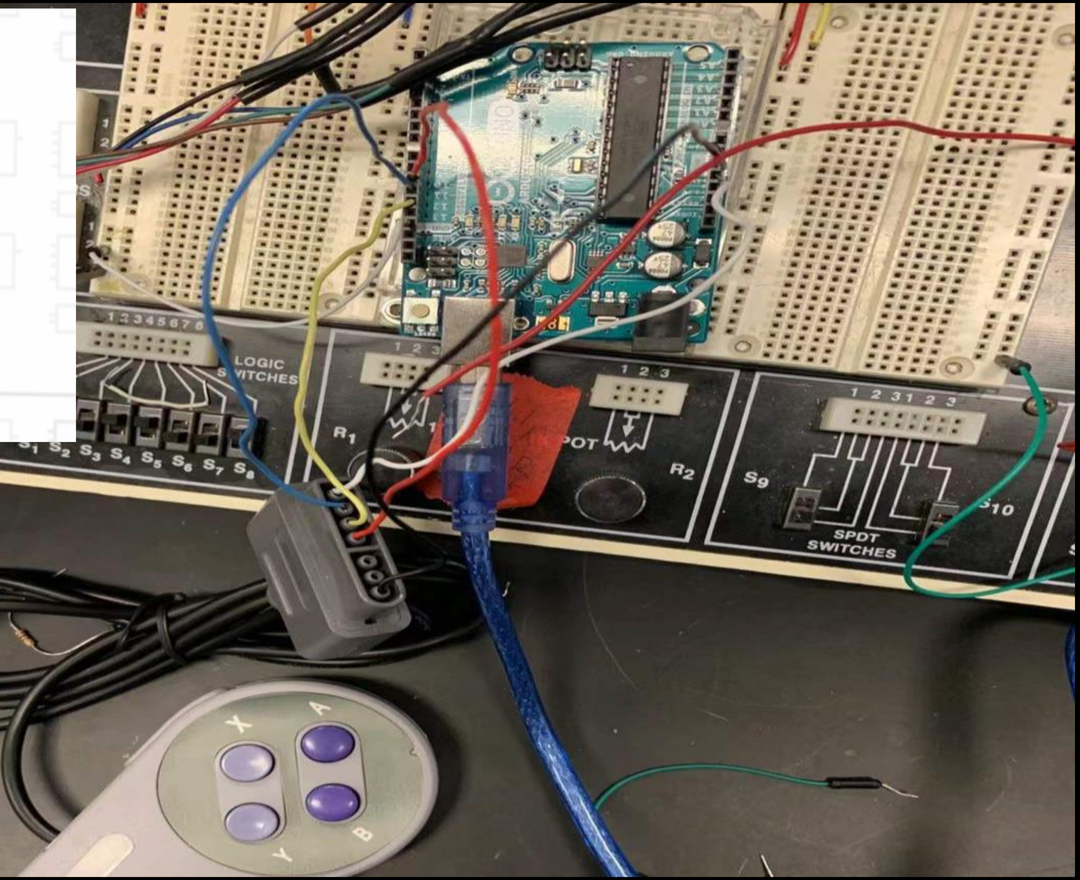


```
int noteDurations[] = { 4, 8, 8, 4, 4, 4, 4, 4
};

int buzzer = 8;
int Switch = 1;
void setup() {
  pinMode(10, INPUT);
  // iterate over the notes of the melody:
  Serial.begin(9600);
}

void loop() {
  Switch = digitalRead(10);
Serial.println(Switch);

  if (Switch == 0) {// input signal
    for (int Note = 0; Note < 8; Note++) {

      // to calculate the note duration, take one second divided by the note type.

      int noteDuration = 1000 / noteDurations[Note];

      tone(buzzer, melody[Note], noteDuration);

      int pause = noteDuration * 1.30; // set a minimum time between them. note's duration + 30%

      delay(pause);

      noTone(buzzer);     // stop the tone playing:

    }
  }
```

SNES Controller

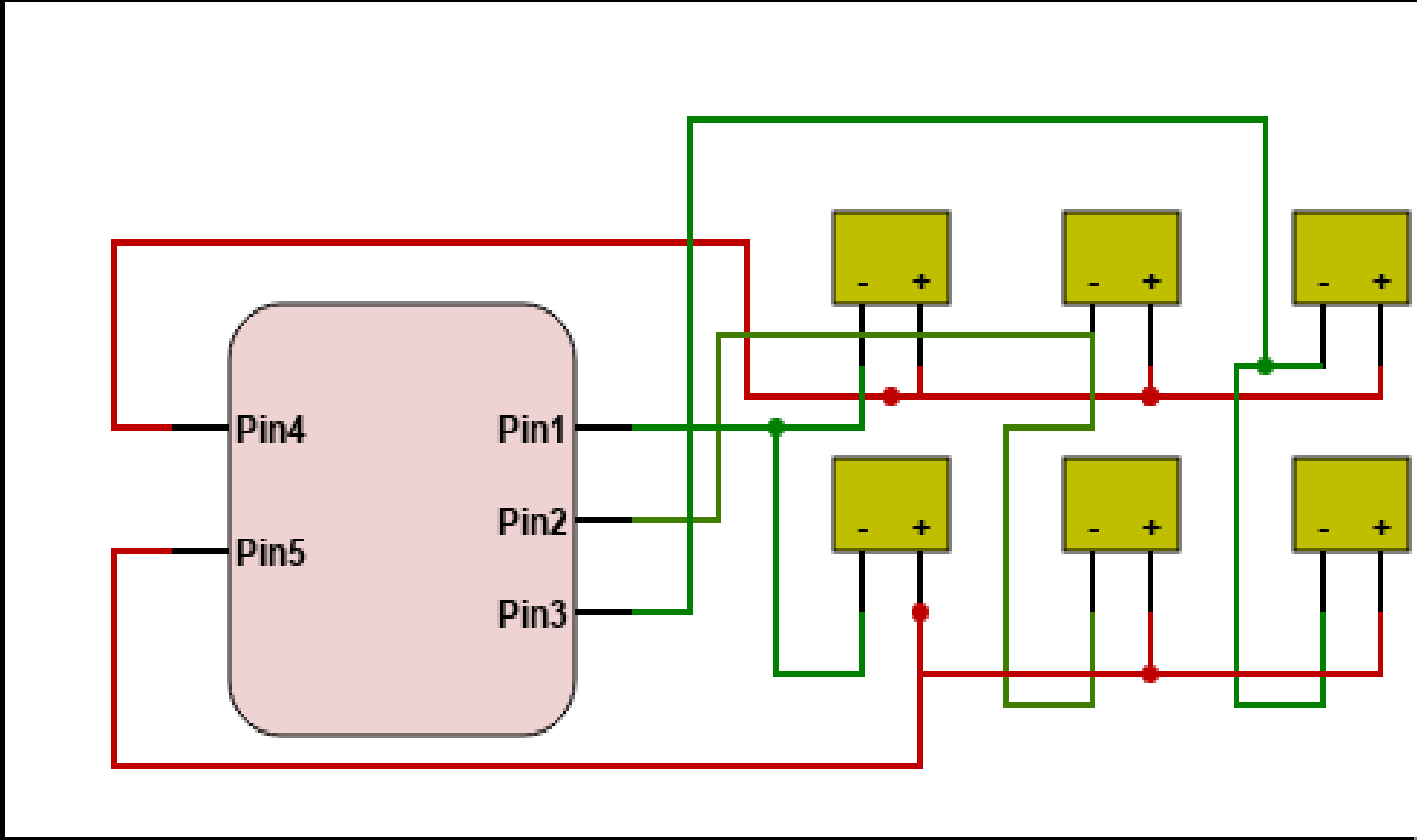


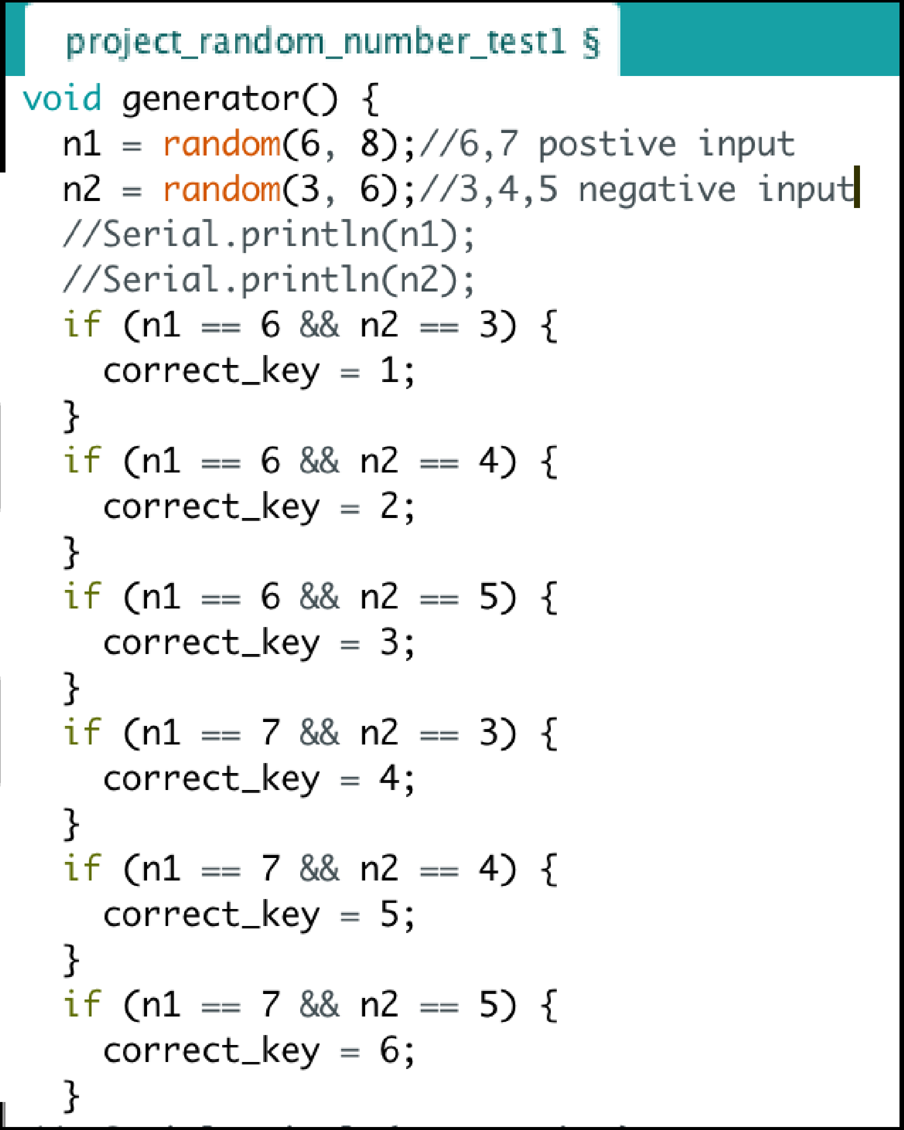




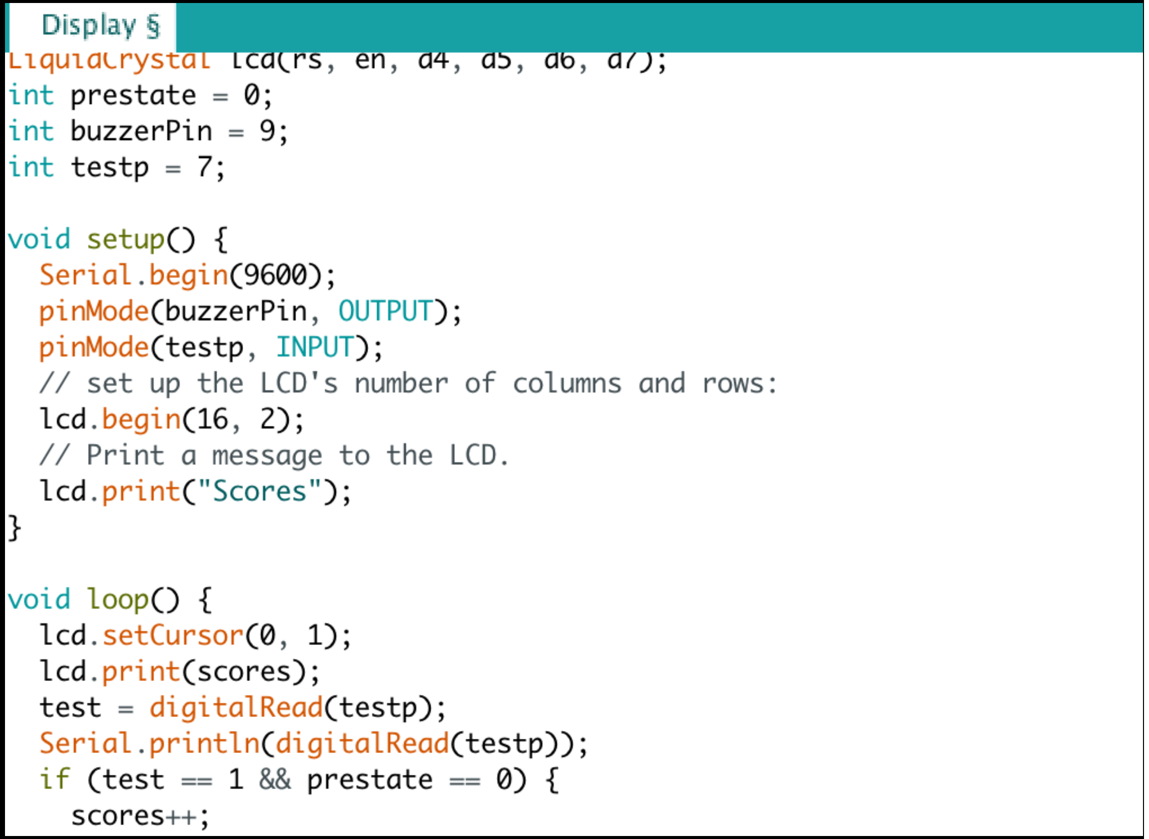
	1	2	3	4	5	6	7	8	9	10	11	12
Left						0						
Up				0								
Right							0					
Down					0							
Y	0											
X									0			
A								0				
Select		0										
Start			0									
L										0		
R											0	
B												0

3X3 Matrix LEDs

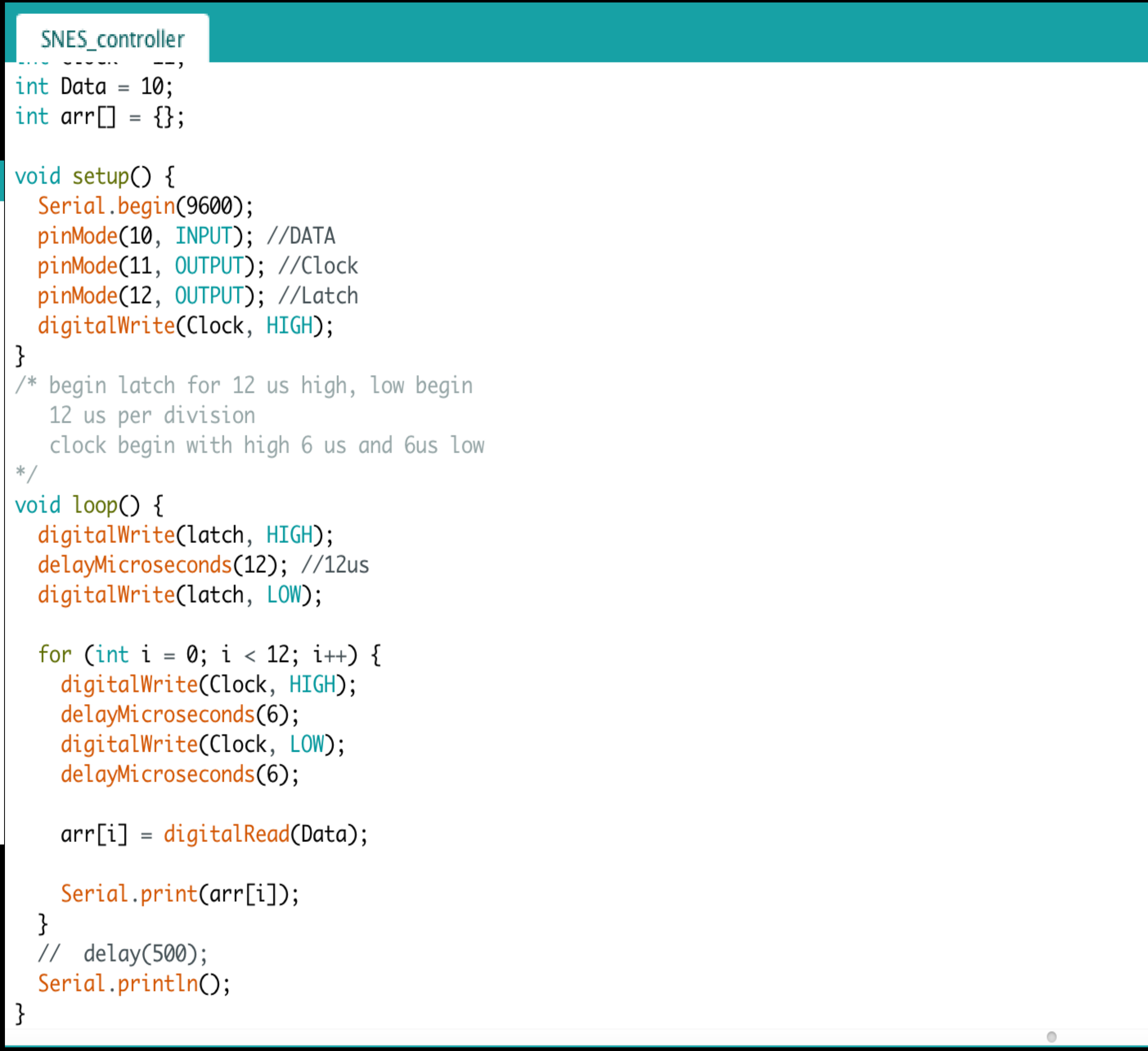




```
void generator() {
  n1 = random(6, 8);//6,7 postive input
  n2 = random(3, 6);//3,4,5 negative input
  //Serial.println(n1);
  //Serial.println(n2);
  if (n1 == 6 && n2 == 3) {
    correct_key = 1;
  }
  if (n1 == 6 && n2 == 4) {
    correct_key = 2;
  }
  if (n1 == 6 && n2 == 5) {
    correct_key = 3;
  }
  if (n1 == 7 && n2 == 3) {
    correct_key = 4;
  }
  if (n1 == 7 && n2 == 4) {
    correct_key = 5;
  }
  if (n1 == 7 && n2 == 5) {
    correct_key = 6;
  }
```



```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
int prestate = 0;
int buzzerPin = 9;
int testp = 7;

void setup() {
  Serial.begin(9600);
  pinMode(buzzerPin, OUTPUT);
  pinMode(testp, INPUT);
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("Scores");
}

void loop() {
  lcd.setCursor(0, 1);
  lcd.print(scores);
  test = digitalRead(testp);
  Serial.println(digitalRead(testp));
  if (test == 1 && prestate == 0) {
    scores++;
```



```
int Data = 10;
int arr[] = {};

void setup() {
  Serial.begin(9600);
  pinMode(10, INPUT); //DATA
  pinMode(11, OUTPUT); //Clock
  pinMode(12, OUTPUT); //Latch
  digitalWrite(Clock, HIGH);
}
/* begin latch for 12 us high, low begin
   12 us per division
   clock begin with high 6 us and 6us low
*/
void loop() {
  digitalWrite(latch, HIGH);
  delayMicroseconds(12); //12us
  digitalWrite(latch, LOW);

  for (int i = 0; i < 12; i++) {
    digitalWrite(Clock, HIGH);
    delayMicroseconds(6);
    digitalWrite(Clock, LOW);
    delayMicroseconds(6);

    arr[i] = digitalRead(Data);

    Serial.print(arr[i]);
  }
  //  delay(500);
  Serial.println();
}
```

Parts

- 8 LEDs
- 8 Debounce Buttons
- 2 Buzzers
- 2 Breadboards, 1 main 1 as controller
- An Arduino Uno micro-controller
- A SNES controller
- An Arduino LCD Display that displays 3 digits