

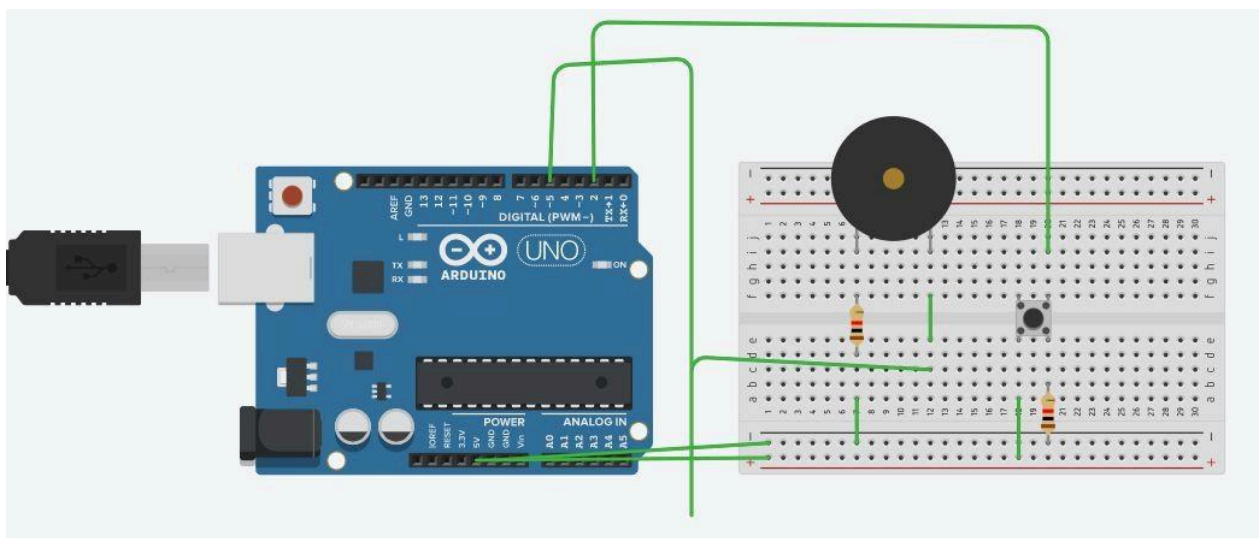
COMP 1045 Lab 7

Student Name: Santiago Cruz

Student ID: 200540981

Date: 03/06/2023

Circuit diagram: Today we will be using the buzzer connected to pin 5 and a button connected to pin 2.



Level 1: Copy and paste the source code to check if your buzzer works.

```
int buzzerPin = 5 ; //The buzzerPin is connected to pin 5 of the Arduino.
int button1Pin = 2; //The SW1 button is connect to pin 2 of the Arduino.

void setup() { //The Setup function runs once.
  pinMode(buzzerPin, OUTPUT); //Setup red LED pin as an output pin.
  pinMode(button1Pin, INPUT); //Setup button1 pin as an input pin.
}

void loop() { //The loop function runs forever.
  if (digitalRead(button1Pin) == HIGH) { //Check to see if button1 is
    pressed.
    tone(buzzerPin, 1000,50); //Play a tone of 1000Hz for 50 milliseconds.
  }
}
```

Level 2: We will create a library, in the online version just copy and paste at top of code.

```
/* ****  
* Public Constants  
* **** */  
  
#define NOTE_B0 31  
#define NOTE_C1 33  
#define NOTE_CS1 35  
#define NOTE_D1 37  
#define NOTE_DS1 39  
#define NOTE_E1 41  
#define NOTE_F1 44  
#define NOTE_FS1 46  
#define NOTE_G1 49  
#define NOTE_GS1 52  
#define NOTE_A1 55  
#define NOTE_AS1 58  
#define NOTE_B1 62  
#define NOTE_C2 65  
#define NOTE_CS2 69  
#define NOTE_D2 73  
#define NOTE_DS2 78  
#define NOTE_E2 82  
#define NOTE_F2 87  
#define NOTE_FS2 93  
#define NOTE_G2 98  
#define NOTE_GS2 104  
#define NOTE_A2 110  
#define NOTE_AS2 117  
#define NOTE_B2 123  
#define NOTE_C3 131  
#define NOTE_CS3 139  
#define NOTE_D3 147  
#define NOTE_DS3 156  
#define NOTE_E3 165  
#define NOTE_F3 175  
#define NOTE_FS3 185  
#define NOTE_G3 196
```

```
#define NOTE_GS3 208
#define NOTE_A3 220
#define NOTE_AS3 233

#define NOTE_B3 247
#define NOTE_C4 262
#define NOTE_CS4 277
#define NOTE_D4 294
#define NOTE_DS4 311
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_FS4 370
#define NOTE_G4 392
#define NOTE_GS4 415
#define NOTE_A4 440
#define NOTE_AS4 466
#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_CS5 554
#define NOTE_D5 587
#define NOTE_DS5 622
#define NOTE_E5 659
#define NOTE_F5 698
#define NOTE_FS5 740
#define NOTE_G5 784
#define NOTE_GS5 831
#define NOTE_A5 880
#define NOTE_AS5 932
#define NOTE_B5 988
#define NOTE_C6 1047
#define NOTE_CS6 1109
#define NOTE_D6 1175
#define NOTE_DS6 1245
#define NOTE_E6 1319
#define NOTE_F6 1397
#define NOTE_FS6 1480
#define NOTE_G6 1568
#define NOTE_GS6 1661
```

```
#define NOTE_A6 1760
#define NOTE_AS6 1865
#define NOTE_B6 1976
#define NOTE_C7 2093
#define NOTE_CS7 2217
#define NOTE_D7 2349
#define NOTE_DS7 2489
#define NOTE_E7 2637
#define NOTE_F7 2794

#define NOTE_FS7 2960
#define NOTE_G7 3136
#define NOTE_GS7 3322
#define NOTE_A7 3520
#define NOTE_AS7 3729
#define NOTE_B7 3951
#define NOTE_C8 4186
#define NOTE_CS8 4435
#define NOTE_D8 4699
#define NOTE_DS8 4978
```

Level 2 continued: Copy the following code inside the main loop:

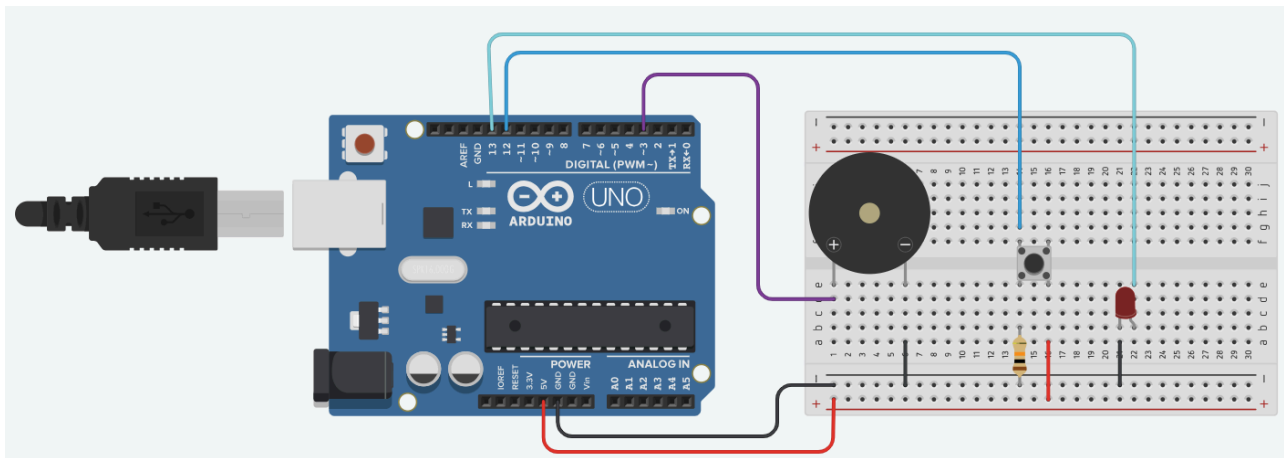
```
int buzzerPin = 5 ; //The buzzerPin is connected to pin 5 of the Arduino.
int button1Pin = 2; //The SW1 button is connect to pin 2 of the Arduino.

void setup() { //The Setup function runs once.
pinMode(buzzerPin, OUTPUT); //Setup red LED pin as an output pin.
pinMode(button1Pin, INPUT); //Setup button1 pin as an input pin.
}

void loop() { //The loop function runs forever.
if (digitalRead(button1Pin) == HIGH) { //Check to see if button1 is
pressed.
tone(buzzerPin, 1000,50); //Play a tone of 1000Hz for 50 milliseconds.
tone(buzzerPin, NOTE_B4,408);
delay(408);
tone(buzzerPin, NOTE_A4,408); delay(408);
tone(buzzerPin, NOTE_G4,408); delay(408);
tone(buzzerPin, NOTE_A4,408); delay(408);
tone(buzzerPin, NOTE_B4,408); delay(408);
```

```
tone(buzzerPin, NOTE_B4,408); delay(408);
tone(buzzerPin, NOTE_B4,408);
}
}
```

Level 3: Create or find a song online and use arrays to store the notes and durations. Then add a light show. The lights can be linked to a specific tone or you can just make a random color.



```
#define NOTE_B0 31
#define NOTE_C1 33
#define NOTE_CS1 35
#define NOTE_D1 37
#define NOTE_DS1 39
#define NOTE_E1 41
#define NOTE_F1 44
#define NOTE_FS1 46
#define NOTE_G1 49
#define NOTE_GS1 52
#define NOTE_A1 55
#define NOTE_AS1 58
#define NOTE_B1 62
#define NOTE_C2 65
#define NOTE_CS2 69
#define NOTE_D2 73
#define NOTE_DS2 78
#define NOTE_E2 82
#define NOTE_F2 87
#define NOTE_FS2 93
#define NOTE_G2 98
```

```
#define NOTE_GS2 104
#define NOTE_A2 110
#define NOTE_AS2 117
#define NOTE_B2 123
#define NOTE_C3 131
#define NOTE_CS3 139
#define NOTE_D3 147
#define NOTE_DS3 156
#define NOTE_E3 165
#define NOTE_F3 175
#define NOTE_FS3 185
#define NOTE_G3 196
#define NOTE_GS3 208
#define NOTE_A3 220
#define NOTE_AS3 233
#define NOTE_B3 247
#define NOTE_C4 262
#define NOTE_CS4 277
#define NOTE_D4 294
#define NOTE_DS4 311
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_FS4 370
#define NOTE_G4 392
#define NOTE_GS4 415
#define NOTE_A4 440
#define NOTE_AS4 466
#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_CS5 554
#define NOTE_D5 587
#define NOTE_DS5 622
#define NOTE_E5 659
#define NOTE_F5 698
#define NOTE_FS5 740
#define NOTE_G5 784
#define NOTE_GS5 831
#define NOTE_A5 880
```

```
#define NOTE_AS5 932
#define NOTE_B5 988
#define NOTE_C6 1047
#define NOTE_CS6 1109
#define NOTE_D6 1175
#define NOTE_DS6 1245
#define NOTE_E6 1319
#define NOTE_F6 1397
#define NOTE_FS6 1480
#define NOTE_G6 1568
#define NOTE_GS6 1661
#define NOTE_A6 1760
#define NOTE_AS6 1865
#define NOTE_B6 1976
#define NOTE_C7 2093
#define NOTE_CS7 2217
#define NOTE_D7 2349
#define NOTE_DS7 2489
#define NOTE_E7 2637
#define NOTE_F7 2794
#define NOTE_FS7 2960
#define NOTE_G7 3136
#define NOTE_GS7 3322
#define NOTE_A7 3520
#define NOTE_AS7 3729
#define NOTE_B7 3951
#define NOTE_C8 4186
#define NOTE_CS8 4435
#define NOTE_D8 4699
#define NOTE_DS8 4978

#define melodyPin 3
// Song of Time notes
int melody[] = {
    NOTE_A4, NOTE_D4, NOTE_F4,
    NOTE_A4, NOTE_D4, NOTE_F4,
    NOTE_A4, NOTE_C5, NOTE_B4, NOTE_G4,
    NOTE_F4, NOTE_G4, NOTE_A4, NOTE_D4,
```

```

    NOTE_C4, NOTE_E4, NOTE_D4
};

// Song of Time tempo
int tempo[] = {
    4, 2, 4,
    4, 2, 4,
    4, 8, 8, 4,
    4, 8, 8, 4,
    4, 8, 1
};

void setup(void)
{
    pinMode(3, OUTPUT); // Buzzer
    pinMode(13, OUTPUT); // Led indicator when singing a note
    pinMode(12, INPUT); // Button
}

void loop()
{
    bool buttonState = digitalRead(12);
    if (buttonState == HIGH) {
        sing(1);
    };
    delay(5);
}

int song = 0;

void sing(int s) {
    // Iterate over the notes of the melody:
    song = s;
    Serial.println("Song of Time");
    int size = sizeof(melody) / sizeof(int);
    for (int thisNote = 0; thisNote < size; thisNote++) {

        int noteDuration = 1500 / tempo[thisNote];

        buzz(melodyPin, melody[thisNote], noteDuration);

```



```

    int pauseBetweenNotes = noteDuration * 0.25;
    delay(pauseBetweenNotes);

    // Stop the tone playing:
    buzz(melodyPin, 0, noteDuration);
}
}

void buzz(int targetPin, long frequency, long length) {
    digitalWrite(13, HIGH);

    long delayValue = 1000000 / frequency / 2; // Calculate the delay value
between transitions

    long numCycles = frequency * length / 1000; // Calculate the number of
cycles for proper timing

    for (long i = 0; i < numCycles; i++) { // For the calculated length of
time...
        digitalWrite(targetPin, HIGH); // Write the buzzer pin high to push out
the diaphragm

        delayMicroseconds(delayValue); // Wait for the calculated delay value

        digitalWrite(targetPin, LOW); // Write the buzzer pin low to pull back
the diaphragm

        delayMicroseconds(delayValue); // Wait again or the calculated delay
value
    }
    digitalWrite(13, LOW);
}

```