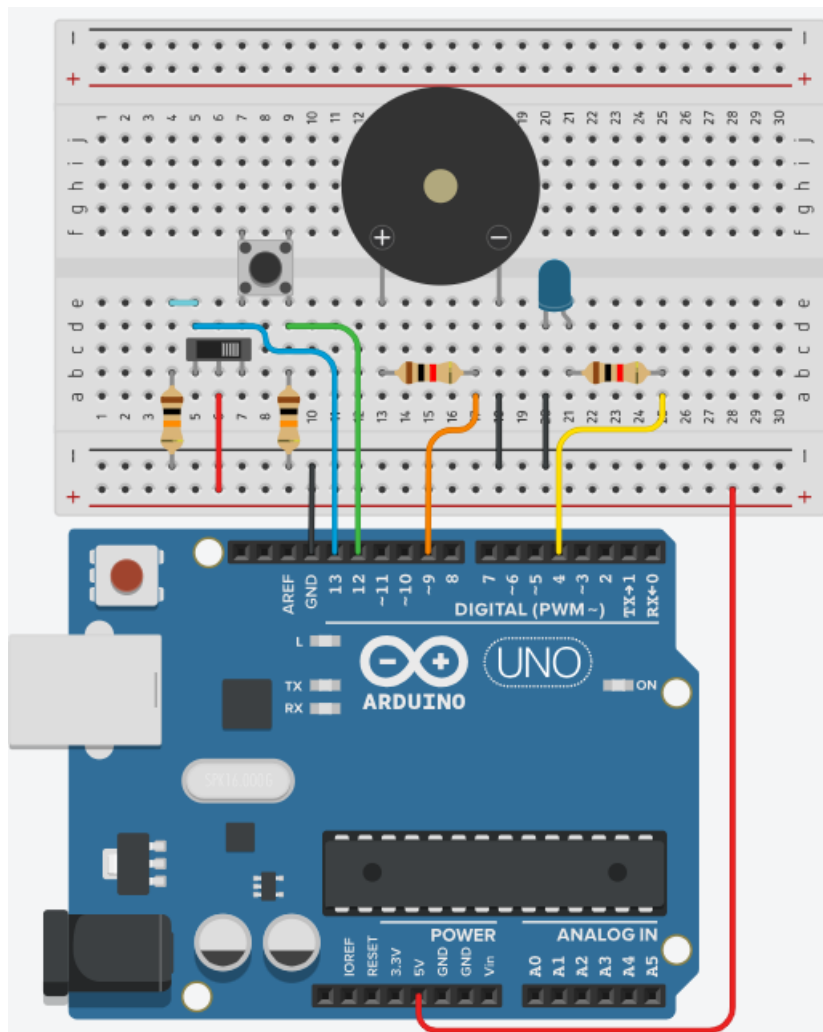




UNIVERSIDADE
DE ÉVORA

08/10/2024



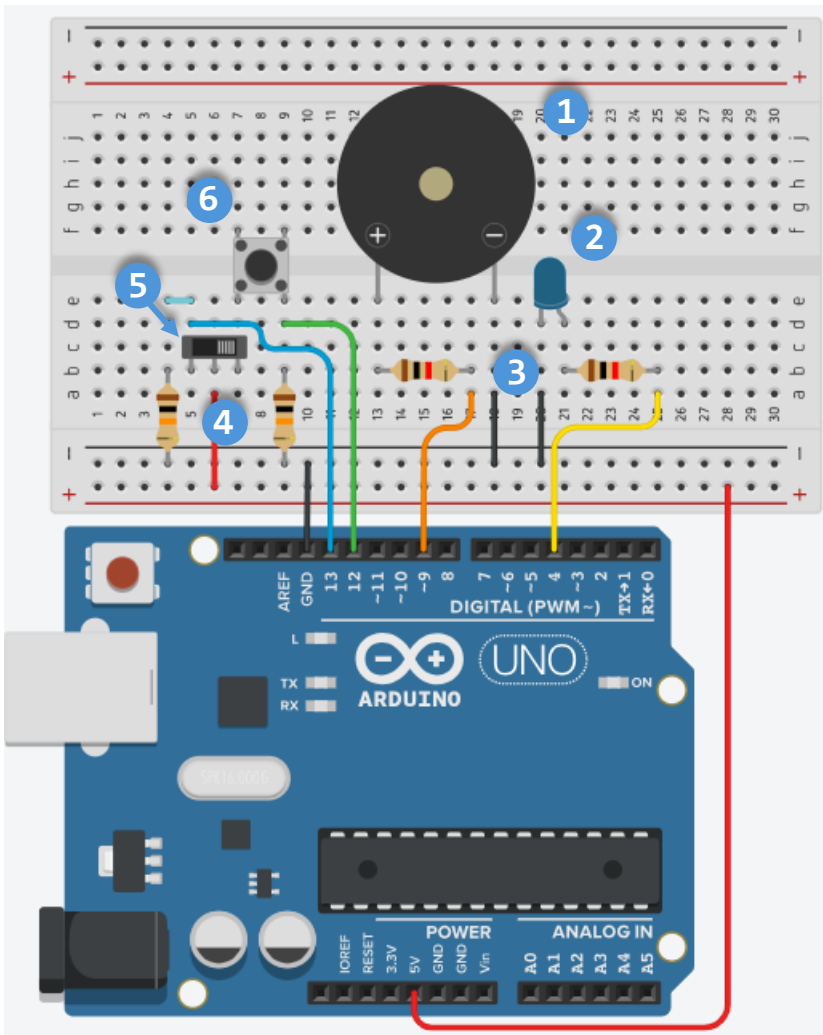
PROJECTO ARDUINO

GRUPO F

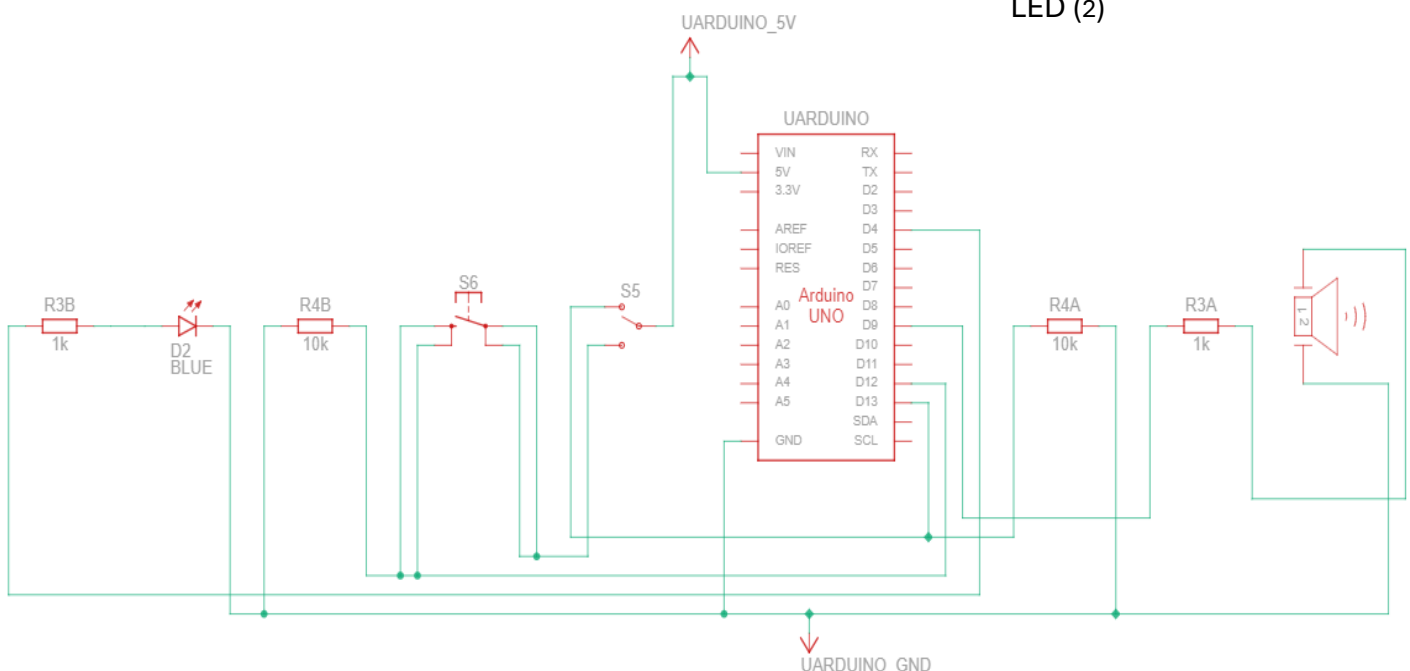
OSCAR BARROS-NOGUEIRA

RAFAELA FERNANDES GROBERIO

Descrição do Hardware Utilizado



- 1 Buzzer**
 - Usado para tocar um tom de 3kHz e um tom de 6kHz
 - Conectado ao PIN 9 no Arduino
- 2 LED Azul**
 - Usado para significar quando o botão (6) não esta a ser pressionado
 - Conectado ao PIN 4 no Arduino
- 3 2x Resistências 1kΩ**
 - Usado para limitar a corrente ao LED (2) e ao Buzzer (1)
- 4 2x Resistências Pull-Down 10kΩ**
 - Usado para dar um valor de 0v (Low) ao Arduino quando o Botão (6) não esta a ser pressionado e o Botão Deslizante (5) esta no estado ligado
- 5 Botão Deslizante**
 - Usado para ligar e desligar a voltagem positiva ao Botão (6), e para controlar se o código associado com o LED (2) deveria correr ou não
- 6 Botão**
 - Usado para controlar o código associado com o Buzzer (1) e o LED (2)



Código

```
4 // Define pins
5 const int LED = 4;
6 const int BUZZER = 9;
7 const int BUTTON = 12;
8 const int SWITCH = 13;
9
10 // Define input/output pins
11 void setup()
12 {
13   pinMode(LED, OUTPUT);
14   pinMode(BUZZER, OUTPUT);
15   pinMode(BUTTON, INPUT);
16   pinMode(SWITCH, INPUT);
17 }
18
19 void loop()
20 {
21   int button_value = digitalRead(BUTTON);           //ler valor do button
22   int switch_value = digitalRead(SWITCH);           //ler valor do switch
23
24   if (switch_value == LOW){                         //verifica se switch está ligado
25     if (button_value == HIGH){                     //verifica se button está pressionado
26       for (int buzzer_loop = 0; buzzer_loop <= 3; buzzer_loop++){ //loop 4x
27         tone(BUZZER, 3000);                         //tocar 3kHz no buzzer
28         delay(1000);                                //esperar 1 segundo
29         tone(BUZZER, 6000);                         //tocar 6kHz no buzzer
30         delay(1000);                                //esperar 1 segundo
31       }
32       noTone(BUZZER);                               //desligar o buzzer
33     } else if (button_value == LOW){                 //verificar se button não está pressionado
34       for (int led_loop = 0; led_loop <= 3; led_loop++){ //Loop 4x
35         digitalWrite(LED, HIGH);                   //ligar LED
36         delay(500);                                //esperar 0.5 segundos
37         digitalWrite(LED, LOW);                    //desligar LED
38         delay(500);                                //esperar 0.5 segundos
39       }
40       delay(2000);                                  //esperar 2 segundos
41     }
42   }
43 }
```

Link do Tinkercad

Explicação do Código

```
4 // Define pins
5 const int LED = 4;
6 const int BUZZER = 9;
7 const int BUTTON = 12;
8 const int SWITCH = 13;
```

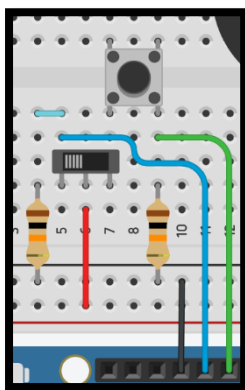
Definir os pins de uma variável constante para facilitar na visualização dos componentes associados com os pins para o uso nos códigos.

Definir quais dos pins são input e output.
Neste caso:

```
10 // Define input/output pins
11 void setup()
12 {
13   pinMode(LED, OUTPUT);
14   pinMode(BUZZER, OUTPUT);
15   pinMode(BUTTON, INPUT);
16   pinMode(SWITCH, INPUT);
17 }
```

- O led (pin 4) é um output porque precisamos ligar e desligar-lho.
- O buzzer (pin 9) é um output porque precisamos controlar a frequência do som.
- O button (pin 12) é input porque iremos precisar saber se ele está pressionado ou não para que o Código principal ocorra.
- O switch (pin 13) é input porque precisamos saber o valor dele para controlar se o código devia andar.

```
24 if (switch_value == LOW){
```



Na maneira que nos temos os componentes eletrônicos conectados, o Arduino está a ler o pin esquerdo do botão deslizante, quando o botão deslizante está para a esquerda não há nenhuma voltagem a passar para o botão, mas há uma voltagem de 5 volts para o pin esquerdo do botão, e significa que o Sistema deveria estar desligado.

No caso oposto (botão quando está para a direita) o Sistema está ligado e há uma corrente de 5 volts a passar para o botão e o valor do switch está 0 volts.

```

25  if (button_value == HIGH){
26      for (int buzzer_loop = 0; buzzer_loop <= 3; buzzer_loop++){
27          tone(BUZZER, 3000);
28          delay(1000);
29          tone(BUZZER, 6000);
30          delay(1000);
31      }
32      noTone(BUZZER);

```

Se o botão estiver ligado, cria uma variável (com valor 0) verifica se a variável é menor ou igual a 3 (para que o Código corra 4 vezes), se é verdade corre o código e adiciona 1 a variável.

Se o for é verdade, então o buzzer toca um tom de 3 kHz por 1 segundo e a seguir toca um tom de 6 kHz por 1 segundo, apos ocorrer 4 vezes ele sai do for e corre o código para parar o som do buzzer.

```

33  } else if (button_value == LOW){
34      for (int led_loop = 0; led_loop <= 3; led_loop++){
35          digitalWrite(LED, HIGH);
36          delay(500);
37          digitalWrite(LED, LOW);
38          delay(500);
39      }
40      delay(2000);

```

Se o botão estiver ligado, cria uma variável, verifica se é menor ou igual a 3, se é verdade corre o código, e adiciona 1 a variável .

Se o for é verdade a led fica ligada por 0.5 segundos e desligado por 0.5 segundos (ele funciona num período de 1 segundo), apos correr 4 vezes ele sai do for e espera 2 segundo e começa o for outra vez.