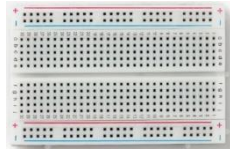


Componentes envolvidos:

Led Vermelho



Breadboard



Resistência



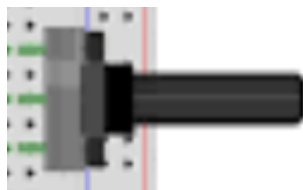
Arduino



Cabos



Potenciômetro

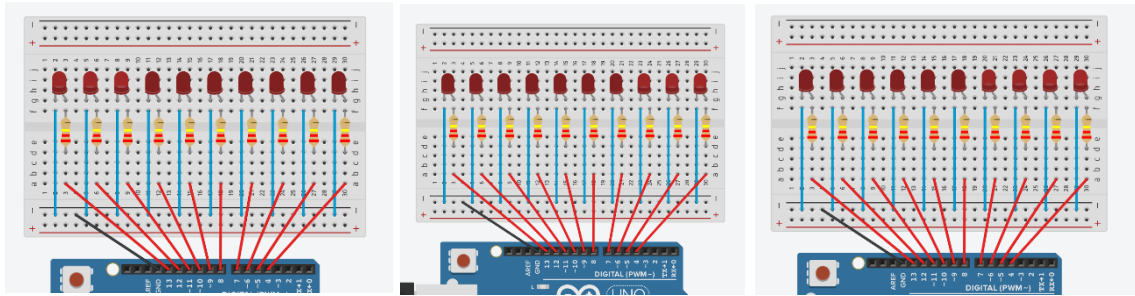


Pin coluna



Exercício 1

Montagem do circuito e respectivos testes:



Código Utilizado:

```
byte ledPin[] = {4, 5, 6, 7, 8, 9, 10, 11, 12, 13};
int ledDelay(65);
int direction = 1;
int currentLED = 0;
unsigned long changeTime;

void setup()
{
  for (int x = 0; x < 10; x++)
  {
    pinMode(ledPin[x], OUTPUT);
  }
  changeTime = millis();
}

void loop()
{
  if((millis() - changeTime) > ledDelay)
  {
    changeLED();
    changeTime = millis();
  }
}
```

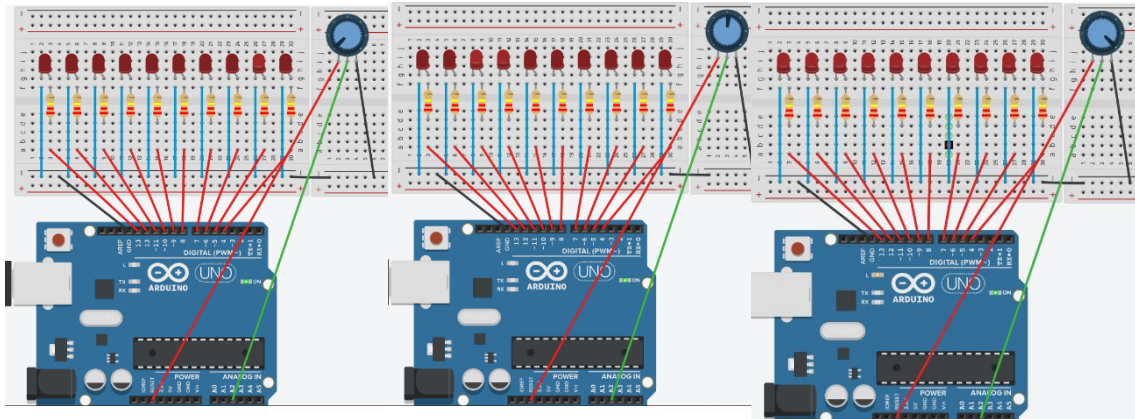
```
void changeLED()
{
  for (int x = 0; x < 10; x++)
  {
    digitalWrite(ledPin[x], LOW);
  }
  digitalWrite(ledPin[currentLED], HIGH);
  currentLED += direction;

  if (currentLED == 9)
  {
    direction = -1;
  }
  if (currentLED == 0)
  {
    direction = 1;
  }
}
```

Neste exercício as luzes vão se ligando sequencialmente, sendo que apos a primeira se ligar, liga se a segunda e desliga se a anterior, assim sucessivamente.

Exercício 2

Montagem do circuito e respectivos testes:



Código Utilizado:

```
byte ledPin[] = {4, 5, 6, 7, 8, 9, 10, 11, 12, 13};

int ledDelay;
int direction = 1;
int currentLED = 0;
unsigned long changeTime;
int potPin = 2;

void setup()
{
  for (int x = 0; x < 10; x++)
  {
    pinMode(ledPin[x], OUTPUT);
  }
  changeTime = millis();
}

void loop()
{
  ledDelay = analogRead(potPin);

  if((millis() - changeTime) > ledDelay)
  {
    changeLED();
    changeTime = millis();
  }
}
```

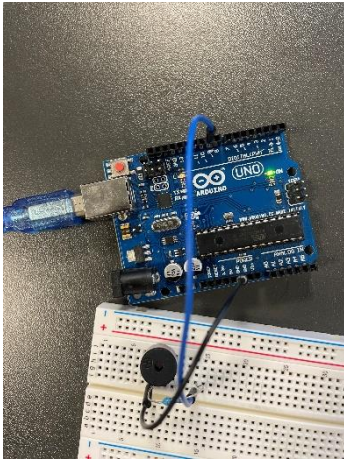
```
void changeLED()
{
  for (int x = 0; x < 10; x++)
  {
    digitalWrite(ledPin[x], LOW);
  }
  digitalWrite(ledPin[currentLED], HIGH);
  currentLED += direction;

  if (currentLED == 9)
  {
    direction = -1;
  }
  if (currentLED == 0)
  {
    direction = 1;
  }
}
```

Neste exercício as luzes vão se ligando sequencialmente conforme os valores recebidos no potenciômetro, sendo que apos a primeira se ligar, liga se a segunda e desliga se a anterior,

Exercício 3

Montagem do circuito e respectivos testes:



Código Utilizado:

```
int speakerPin = 9;

int length = 15;
char notes[] = "ccggaagffeeddc";
int beats[] = {1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 2, 4};
int tempo = 300;

void playTone(int tone, int duration)
{
  for(long i = 0; i < duration * 1000L; i += tone * 2)
  {
    digitalWrite(speakerPin, HIGH);
    delayMicroseconds(tone);
    digitalWrite(speakerPin, LOW);
    delayMicroseconds(tone);
  }
}

void playNote(char note, int duration)
{
  char names[] = {'c', 'd', 'e', 'f', 'g', 'a', 'b', 'C'};
  int tones[] = {1915, 1700, 1519, 1432, 1275, 1136, 1014, 956};
  for (int i = 0; i < 8; i++)
  {
    if(names[i] == note)
    {
      playTone(tones[i], duration);
    }
  }
}
```

```
void setup()
{
  pinMode(speakerPin, OUTPUT);
}

void loop()
{
  for(int i = 0; i < length; i++)
  {
    if(notes[i] == ' ')
    {
      delay(beats[i] * tempo);
    }
    else
    {
      playNote(notes[i], beats[i] * tempo);
    }
    delay(tempo / 2);
  }
}
```

Ao correr este programa o Pin Coluna toca uma música cujas notas e tons estão definidos no `void playNote`.

Exercício 4

Leitura do *Serial Monitor*:

Serial Monitor	
Message (Enter to send message to 'Arduino Uno')	
Adress: 27	Value: 255
Adress: 28	Value: 255
Adress: 29	Value: 255
Adress: 30	Value: 255
Adress: 31	Value: 255
Adress: 32	Value: 255
Adress: 33	Value: 255
Adress: 34	Value: 255
Adress: 35	Value: 255
Adress: 36	Value: 255
Adress: 37	Value: 255
Adress: 38	Value: 255
Adress: 39	Value: 255
Adress: 40	Value: 255
Adress: 41	Value: 255
Adress: 42	Value: 255
Adress: 43	Value: 255

Código Utilizado:

```
#include <EEPROM.h>

int a = 0, i;
int value;

void setup ()
{
  Serial.begin(9600);
}

void loop()
{
  value = EEPROM.read(a);

  Serial.print("Adress: ");
  Serial.print(a);
  Serial.print("\t");
  Serial.print("Value: ");
  Serial.print(value);
  Serial.println();

  a = a + 1;

  if (a == 1024)
  {
    a = 0;
  }
  delay (100);
}
```

Este programa lê o espaço armazenado EEPROM do Arduino e percorre os *address* e diz o espaço em bytes armazenado lá no momento, um por um.

Exercício 5

Leitura do *Serial Monitor*:

Output	Serial Monitor	✕
Message (Enter to send message to 'Arduino Uno')		
Adress: 0	Value: 23	
Adress: 1	Value: 255	
Adress: 2	Value: 255	
Adress: 3	Value: 255	
Adress: 4	Value: 255	
Adress: 5	Value: 255	
Adress: 6	Value: 255	
Adress: 7	Value: 255	
Adress: 8	Value: 255	
Adress: 9	Value: 255	
Adress: 10	Value: 255	

Código Utilizado:

```
#include <EEPROM.h>

int a= 0, i;
int value;

void setup ()
{
  Serial.begin(9600);
  EEPROM.write(0, 23);
}

void loop()
{
  value = EEPROM.read(a);

  Serial.print("Adress: ");
  Serial.print(a);
  Serial.print("\t");
  Serial.print("Value: ");
  Serial.print(value);
  Serial.println();

  a = a + 1;

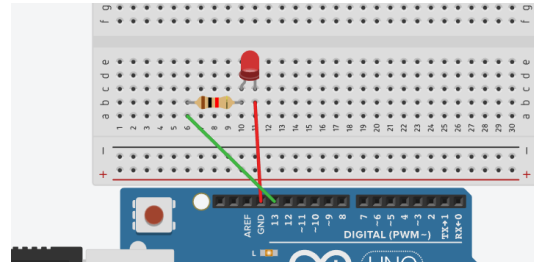
  if (a == 1024)
  {
    a = 0;
  }
  delay (100);
}
```

Em relação ao exercício anterior, o endereço 0 foi atribuído o valor de 23 e nos restantes 255.

Exercício 6

Leitura do *Serial Monitor* e montagem do Arduino:

Output Serial Monitor X			
Message (Enter to send message to 'Arduino Uno' on 'COM5')			
Address: 0	Value: 0	Voltage (V): 0.0	
Address: 1	Value: 31	Voltage (V): 0.155	
Address: 2	Value: 62	Voltage (V): 1.55	
Address: 3	Value: 93	Voltage (V): 1.210	
Address: 4	Value: 124	Voltage (V): 2.110	
Address: 5	Value: 155	Voltage (V): 3.10	
Address: 6	Value: 186	Voltage (V): 3.165	
Address: 7	Value: 217	Voltage (V): 4.65	
Address: 8	Value: 248	Voltage (V): 4.220	
Address: 9	Value: 0	Voltage (V): 0.0	



Código Utilizado:

```
#include <EEPROM.h>

int a = 0;
int value;
int RedPin = 3;

void setup()
{
  Serial.begin(9600);
  EEPROM.write(0, 0);
  EEPROM.write(1, 31);
  EEPROM.write(2, 62);
  EEPROM.write(3, 93);
  EEPROM.write(4, 124);
  EEPROM.write(5, 155);
  EEPROM.write(6, 186);
  EEPROM.write(7, 217);
  EEPROM.write(8, 248);

  pinMode(RedPin, OUTPUT);
}
```

```
void loop()
{
  value = EEPROM.read(a);
  analogWrite(RedPin, value);

  Serial.print("Address: ");
  Serial.print(a);
  Serial.print("\t");
  Serial.print("Value: ");
  Serial.print(value);
  Serial.print("\t");
  Serial.print("Voltage (V): ");
  Serial.print(value * 5 / 255);
  Serial.print(".");
  Serial.println(value * 5 % 255);

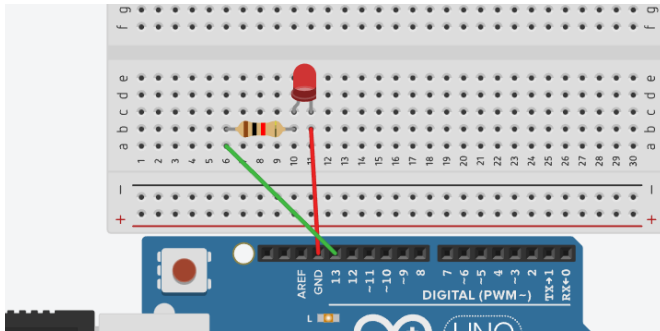
  a = a + 1;
  if(a == 9) a = 0;

  delay(1000);
}
```

Neste exercício é atribuído um valor de intensidade a um LED dependendo do valor em memória, dando um efeito com 9 valores de intensidade num intervalo de 1 segundo.

Exercício 7

Montagem do circuito e respectivos testes:



Código Utilizado:

```
#include<EEPROM.h>

void setup()
{
    pinMode(13, OUTPUT);

    for (int i = 0; i <
EEPROM.length(); i++)
    {
        EEPROM.write(i, 0);
    }
    digitalWrite(13, HIGH);
}

void loop()
{
}
```

Neste programa, no final do ciclo for acende um LED.