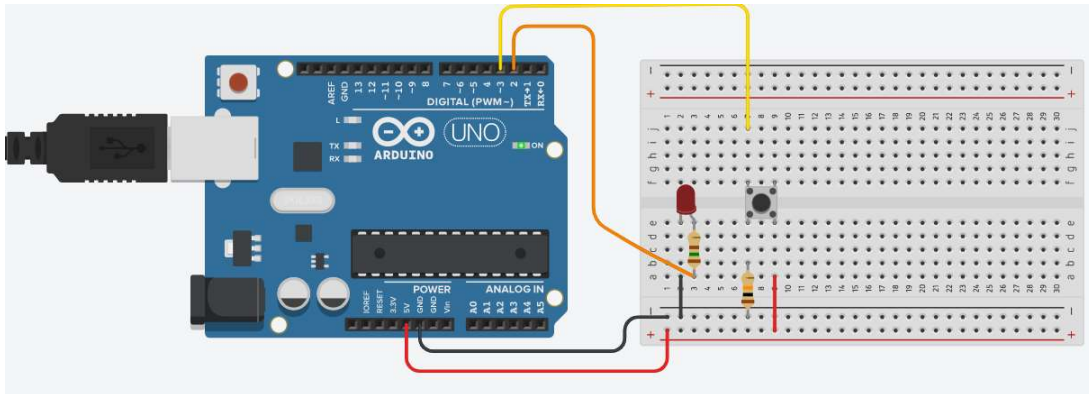
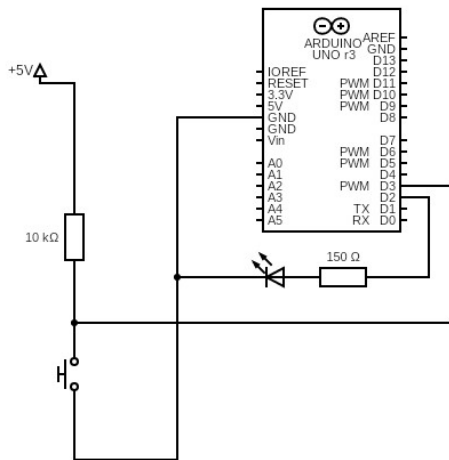


LAB. 4

Alunos: André Silva, Gabriel Duarte e Rui Correia

Exercício 1 (Pull-Down):



```
#define ledPin 2 //Pino do LED
#define butPin 3 //Pino do botão

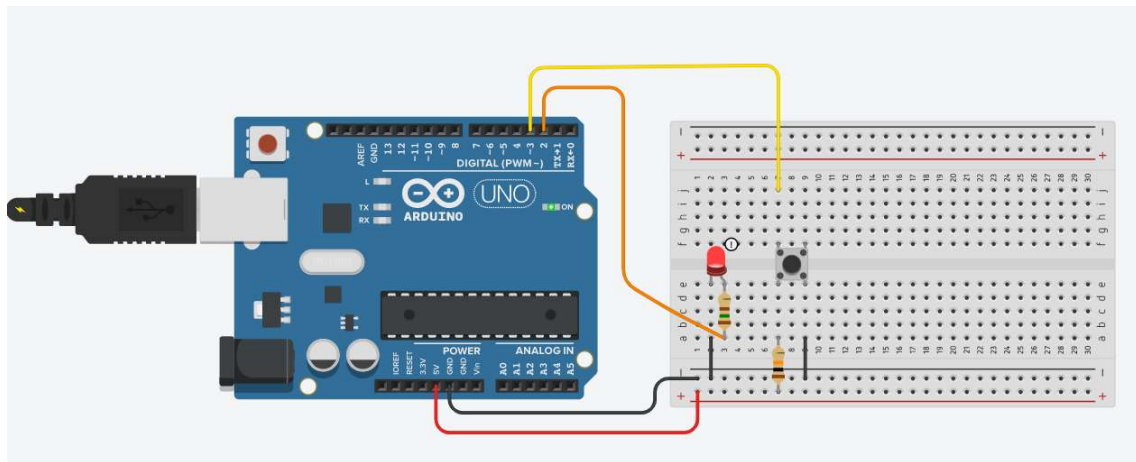
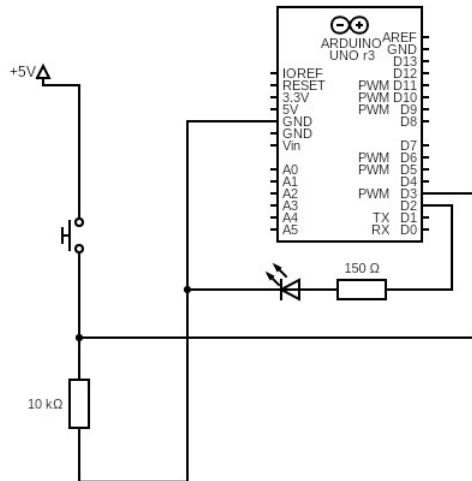
int state = 0; //Variável para o estado do botão

void setup()
{
  pinMode(ledPin, OUTPUT); //Define o pino do LED como OUT
  pinMode(butPin, INPUT); //Define o pino do botão como IN
  Serial.begin(9600); //Taxa de transmissão em bits/s
}

void loop()
{
  state = digitalRead(butPin); //Lê o estado do botão
  Serial.println(state); //Imprime o estado do botão

  //Verifica se o estado do botão é 1 ou 0
  //Se for 1 (não pressionado), acende o LED
  //Se for 0 (pressionado), desliga o LED
  if(state==1){
    digitalWrite(ledPin, HIGH);
  }
  else{
    digitalWrite(ledPin, LOW);
  }
}
```

Exercício 2 (Pull-Up):



```
#define ledPin 2 //Pino do LED
#define butPin 3 //Pino do botão

int state = 0; //Variável para o estado do botão

void setup()
{
  pinMode(ledPin, OUTPUT); //Define o pino do LED como OUT
  pinMode(butPin, INPUT); //Define o pino do botão como IN
  Serial.begin(9600); //Taxa de transmissão em bits/s
}

void loop()
{
  state = digitalRead(butPin); //Lê o estado do botão
  Serial.println(state); //Imprime o estado do botão

  //Verifica se o estado do botão é 1 ou 0
  //Se for 1 (não pressionado), acende o LED
  //Se for 0 (pressionado), desliga o LED
  if(state==1){
    digitalWrite(ledPin, HIGH);
  }
  else{
    digitalWrite(ledPin, LOW);
  }
}
```

Exercício 3:

```

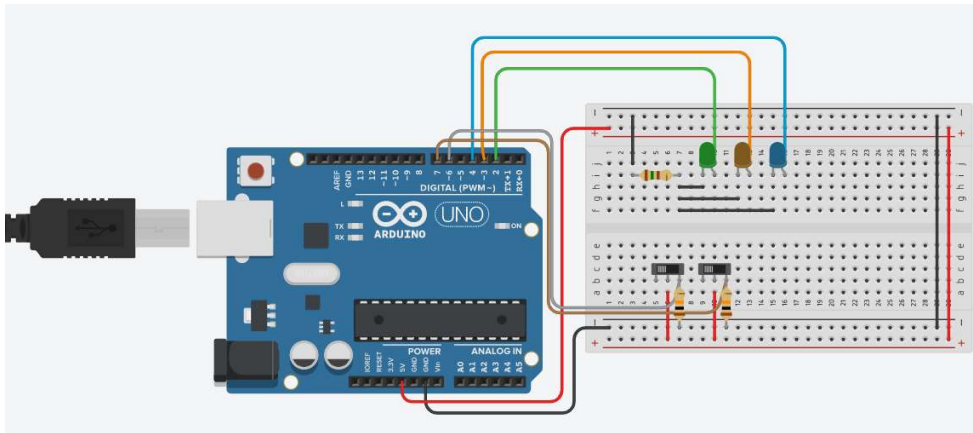
#define ledPin 2 //Pino do LED
#define butPin 3 //Pino do botão

void setup()
{
  pinMode(ledPin, OUTPUT); //Define o pino do LED como OUT
  pinMode(butPin, INPUT); //Define o pino do botão como IN
  Serial.begin(9600); //Taxa de transmissão em bits/s
}

void loop()
{
  //Define o estado do LED em função do estado do botão
  digitalWrite(ledPin, digitalRead(butPin));
}

```

Exercício 4:



```

#define ledAND 2 //Define o led da AND na porta 2
#define ledOR 3 //Define o led da OR na porta 3
#define ledXOR 4 //Define o led da XOR na porta 4

#define but1 6 //Define o botão 1 na porta 6
#define but2 7 //Define o botão 2 na porta 7

int stateBut1 = 0;
int stateBut2 = 0;

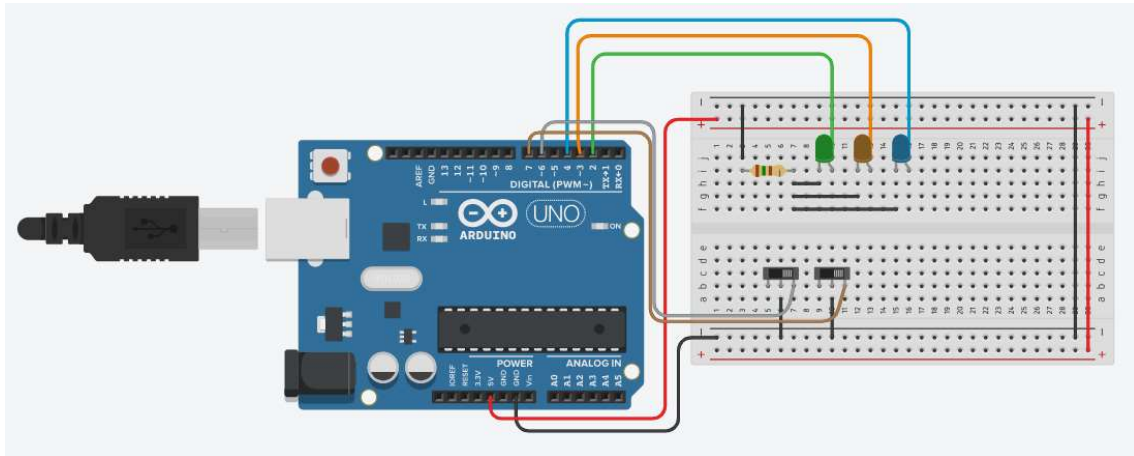
void setup()
{
  //Define os LEDs como saída e os botões como entrada
  pinMode(ledAND, OUTPUT);
  pinMode(ledOR, OUTPUT);
  pinMode(ledXOR, OUTPUT);
  pinMode(but1, INPUT);
  pinMode(but2, INPUT);
}

void loop()
{
  //Lê o estado dos botões
  stateBut1 = digitalRead(but1);
  stateBut2 = digitalRead(but2);

  //Com base nas operações lógicas, acende ou apaga os LEDs
  digitalWrite(ledAND, stateBut1 & stateBut2);
  digitalWrite(ledOR, stateBut1 | stateBut2);
  digitalWrite(ledXOR, stateBut1 ^ stateBut2);
}

```

Exercício Extra:



```
#define ledAND 2 //Define o led da AND na porta 2
#define ledOR 3 //Define o led da OR na porta 3
#define ledXOR 4 //Define o led da XOR na porta 4

#define but1 6 //Define o botão 1 na porta 6
#define but2 7 //Define o botão 2 na porta 7

int stateBut1 = 0;
int stateBut2 = 0;

void setup()
{
    //Define os LEDs como saída e os botões como entrada
    pinMode(ledAND, OUTPUT);
    pinMode(ledOR, OUTPUT);
    pinMode(ledXOR, OUTPUT);
    pinMode(but1, INPUT_PULLUP);
    pinMode(but2, INPUT_PULLUP);
}

void loop()
{
    //Lê e nega o estado dos botões
    stateBut1 = !(digitalRead(but1));
    stateBut2 = !(digitalRead(but2));

    //Com base nas operações lógicas, acende ou apaga os LEDs
    digitalWrite(ledAND, stateBut1 & stateBut2);
    digitalWrite(ledOR, stateBut1 | stateBut2);
    digitalWrite(ledXOR, stateBut1 ^ stateBut2);
}
```