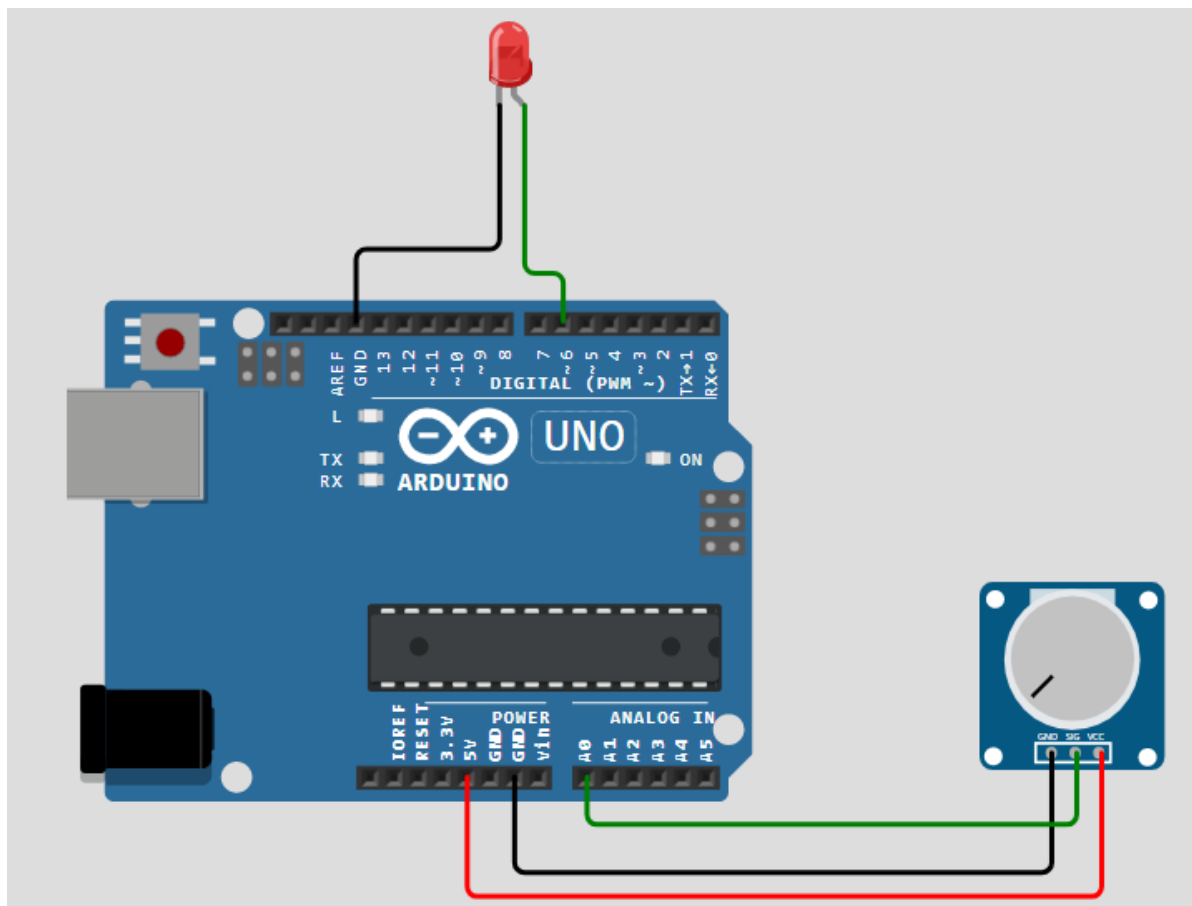


Relatório de Sistemas de Controle – Uso de PWM

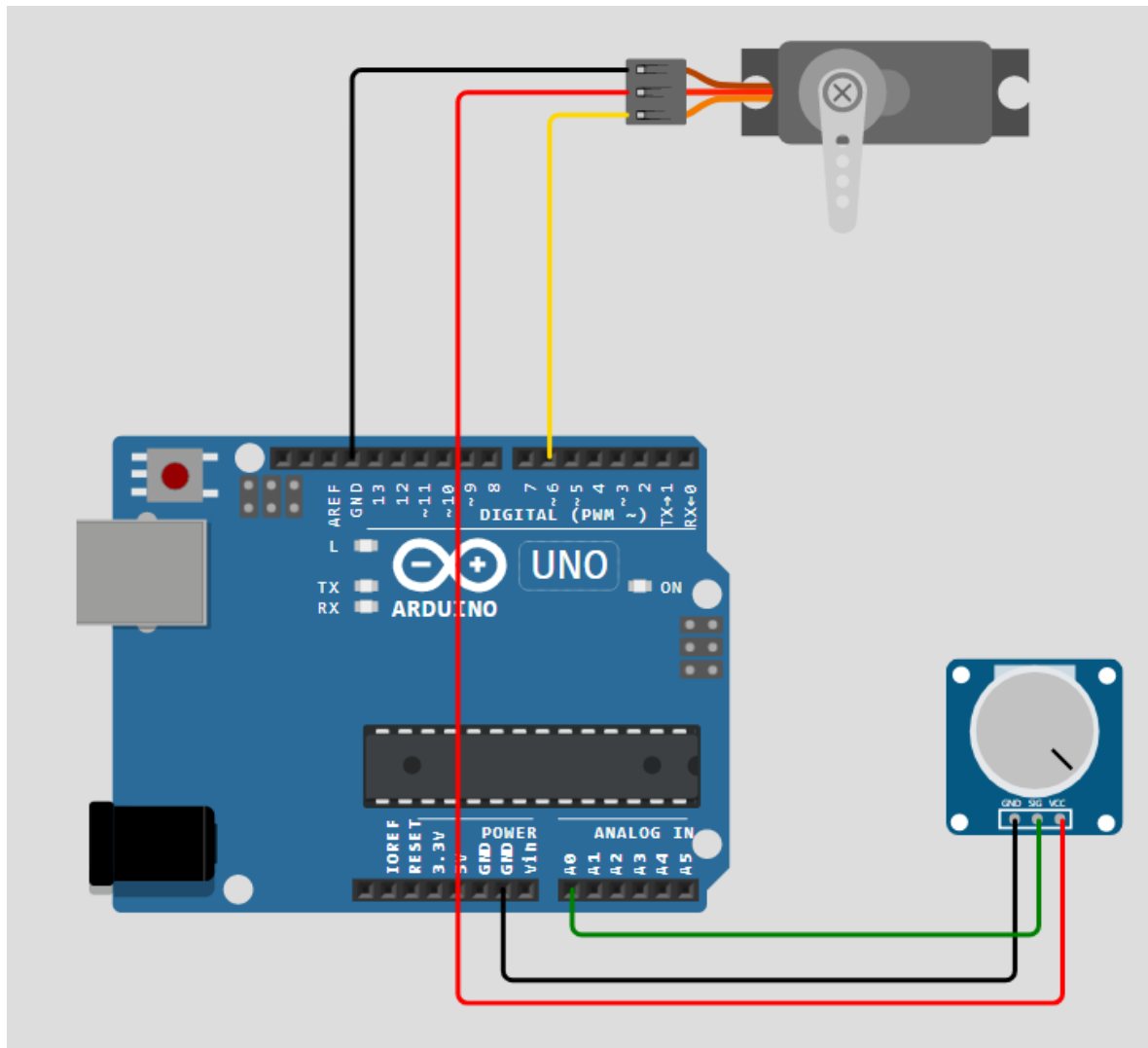
Grupo: Marco Bockoski e Mateus Henrique Cardoso

EXPLICAÇÃO

Desenvolveu-se em sala de aula circuitos que utilizam portas PWM (Portas de modulação por largura de pulso), assim, recebe-se valores analógicos, tendo seus valores alterados pelo uso de potenciômetro e envia-os tanto para LED quanto para um servo-motor para exemplificação de casos. Os códigos são relativamente semelhantes, apenas diferenciando no caso do servo-motor que utiliza uma biblioteca apropriada para seu uso com Arduino. Outra observação é a necessidade de uso de resistor para limitação de corrente sobre o LED.



Aplicação de LED como output do PMW



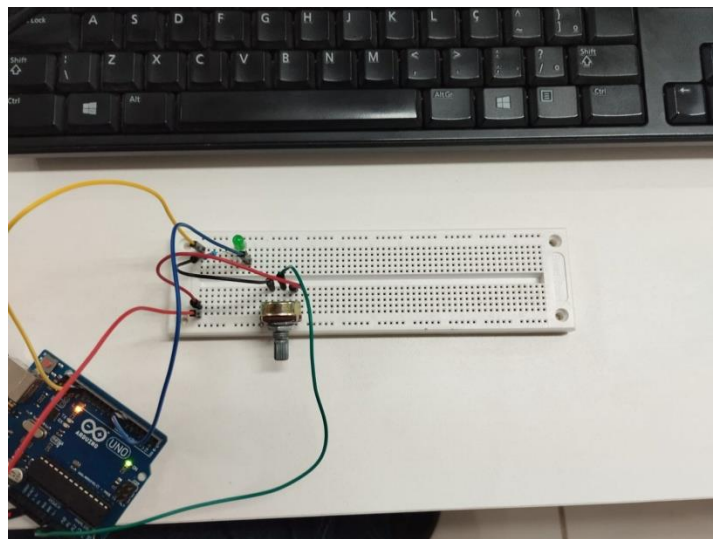
Aplicação do Servo-Motor como Output do PWM

MATERIAIS UTILIZADOS

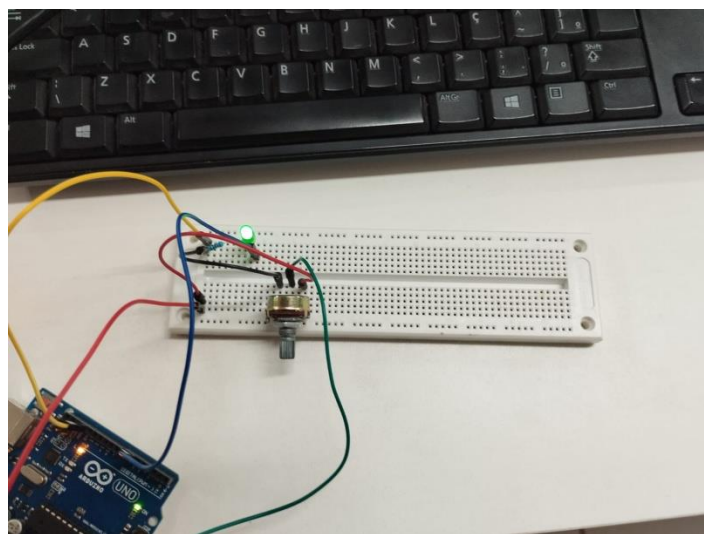
- 8 fios condutores
- 1 cabo USB
- 1 Microcontrolador Arduino Uno
- 1 resistor de 10K
- 1 LED vermelho
- 1 servo-motor
- 1 protoboard
- 1 potenciômetro

MONTAGEM

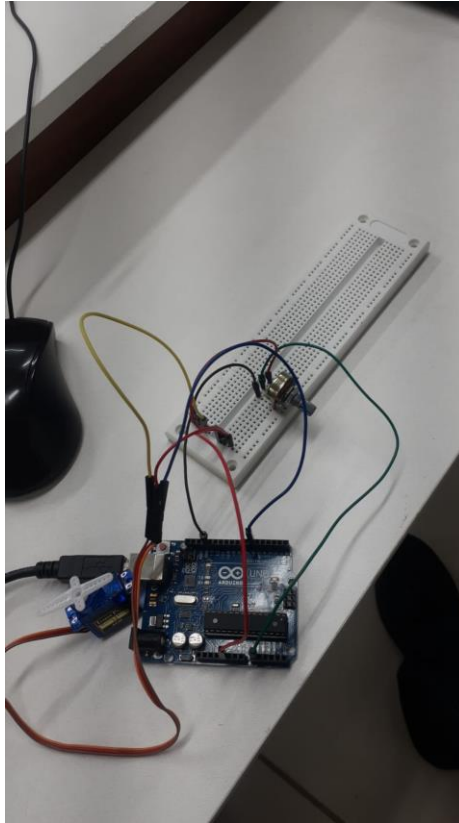
A montagem do circuito com LED estabelece as conexões por meio da protoboard. Dois fios energizam a protoboard por meio da alimentação que o Arduino fornece, dois terminais do potenciômetro recebem alimentação positivo e negativo, enquanto o último terminal (do meio) é ligado à porta analógica A0 do Arduino. O LED é ligado em série ao resistor por meio do catodo, o resistor tem ligação com o GND, enquanto o anodo é ligado ao VCC (5V). Quanto à montagem do servo-motor o processo é bem similar, diferenciando na dispensa do uso de resistor e utilizando os terminais VCC (5V) e o GND para funcionamento apropriado do motor.



Circuito de acionamento do LED (LED OFF)



Circuito de acionamento do LED (LED ON)



Circuito de acionamento do Servo-Motor

CÓDIGO

```
int pinoLed = 6;
int pinoPot = A0;
int valor = 0;

void setup() {
  Serial.begin(9600);
  pinMode(pinoLed, OUTPUT);
}

void loop() {
  valor = analogRead(pinoPot);
  valor = map(valor, 0, 1023, 0, 255);
  Serial.println(valor);
  analogWrite(pinoLed, valor);
}
```

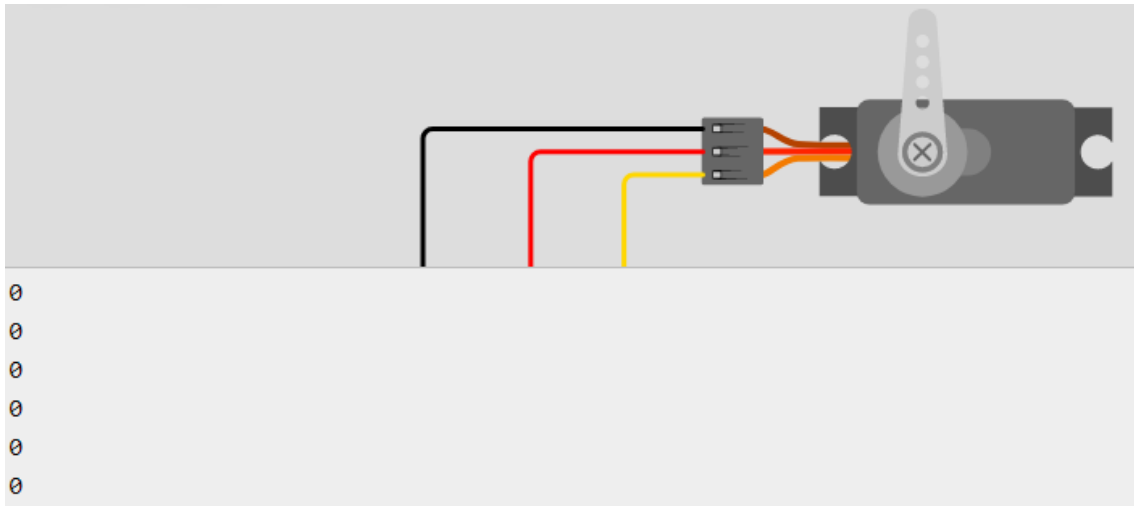
A código acima é aplicável para o acionamento do LED pela porta PWM. Três variáveis são declaradas, uma para receber os valores analógicos do potenciômetro, outro para armazenar o valor da porta conectada ao potenciômetro e a terceira para armazenar o valor da porta do LED, que será declarada como saída (OUTPUT) pelo comando pinMode(), presente na função setup().

Na função loop(), o valor fornecido pelo potenciômetro para a porta A0 por intermédio do comando analogRead(), esse valor é tratado para ser convertido em um input válido para a PWM (Valor máximo 255). Sendo o valor máximo que o potenciômetro fornece 1023, todo valor entre 0 e 1023 lido pelo Arduino é convertido em valores de 0 a 255, proporcionalmente, à porta PWM (6), isso é proporcionado pela função map().

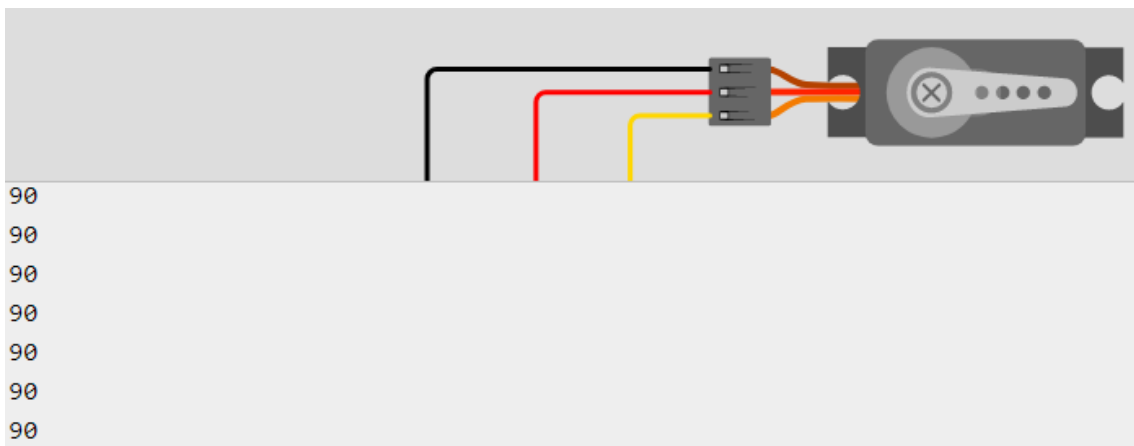
Serial.print é utilizada para aferir valores do potenciômetro e analogWrite fornece os valores para que LED acenda (valor = 255) ou não (valor = 0).

```
1  #include <Servo.h>
2
3  Servo servo;
4  int pinoPot = A0;
5  int valor = 0;
6
7  void setup() {
8      Serial.begin(9600);
9      servo.attach(6);
10 }
11
12 void loop() {
13     valor = analogRead(pinoPot);
14     valor = map(valor, 0, 1023, 0, 180);
15     Serial.println(valor);
16     servo.write(valor);
17     delay(15);
18 }
```

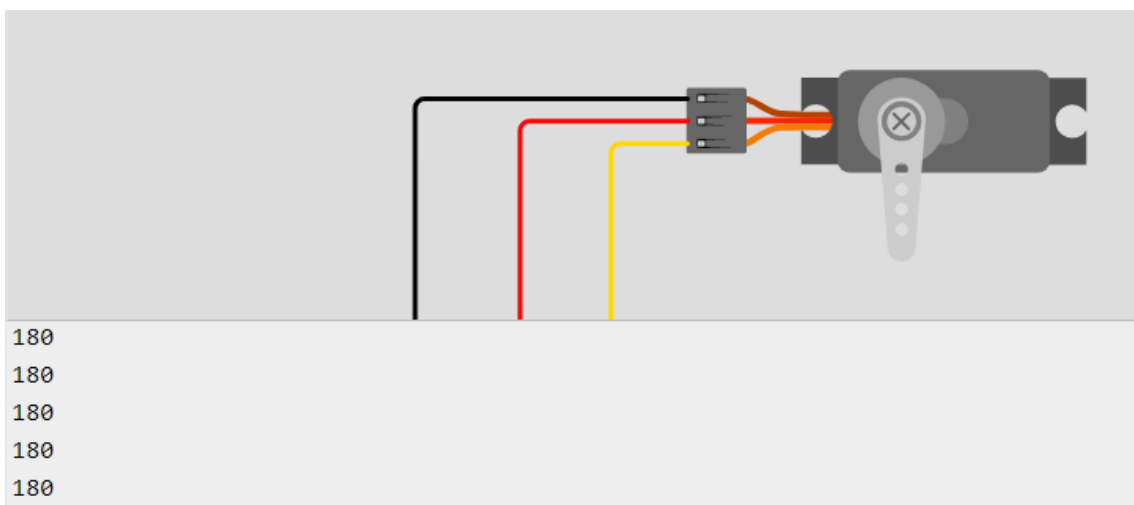
O código acima corresponde ao acionamento do servo-motor, esse script conta com a inclusão da biblioteca Servo.h, que inclui comando referentes ao servo-motor. Na declaração de variáveis, declara-se um objeto "Servo" chamado servo, e o valor da porta PMW utilizada como input do Arduino é definida na linha 9, na qual o comando attach relaciona a porta PMW 6 ao que será utilizada para receber dados e enviar ao servo-motor. O comando map() tem as proporções de 0 a 180 justamente para relacionar ao movimento de 180 graus que o servo-motor realiza:



Servo motor recebendo o input 0 do potenciômetro (posição inicial)



Servo motor recebendo o input 90 do potenciômetro (rotação de 90° em relação a posição inicial)



Servo motor recebendo o input 180 do potenciômetro (rotação de 180° em relação a posição inicial)

Por fim, o comando `servo.write` é o que faz o servo-motor receber o exato valor convertido pelo `map()` para seu funcionamento apropriado.