



Laboratório de Materiais e Equipamentos Elétricos

Roteiro 11: Servo Motor

1. Objetivos

- Analisar e compreender as portas (pinos) de um Arduino.
- Configurar a saída analógica do Arduino para PWM
- Configurar um servo motor

2. Material utilizado

- Potenciômetro de 10k Ohms
- Push-button
- Resistor de 10k Ohms
- Micro Servo SG90
- Protoboard
- Arduino UNO
- Osciloscópio

2.1. Servomotor

Um servomotor é um dispositivo eletromecânico usado para movimentos precisos e controlados. Ele consiste em um motor, um circuito de controle e um mecanismo de feedback. O motor fornece a força necessária para mover o eixo do servomotor. O circuito de controle recebe os sinais de controle e envia comandos para o motor. O mecanismo de feedback, geralmente um potenciômetro, fornece

informações precisas sobre a posição atual do eixo. O servomotor mantém a posição correta, monitorando constantemente o sinal de feedback e fazendo ajustes se necessário. Essa retroalimentação contínua permite que o servomotor seja preciso e estável, mesmo diante de forças externas. Um dos servomotores mais utilizados é o **Micro Servo Motor SG90 9g**.

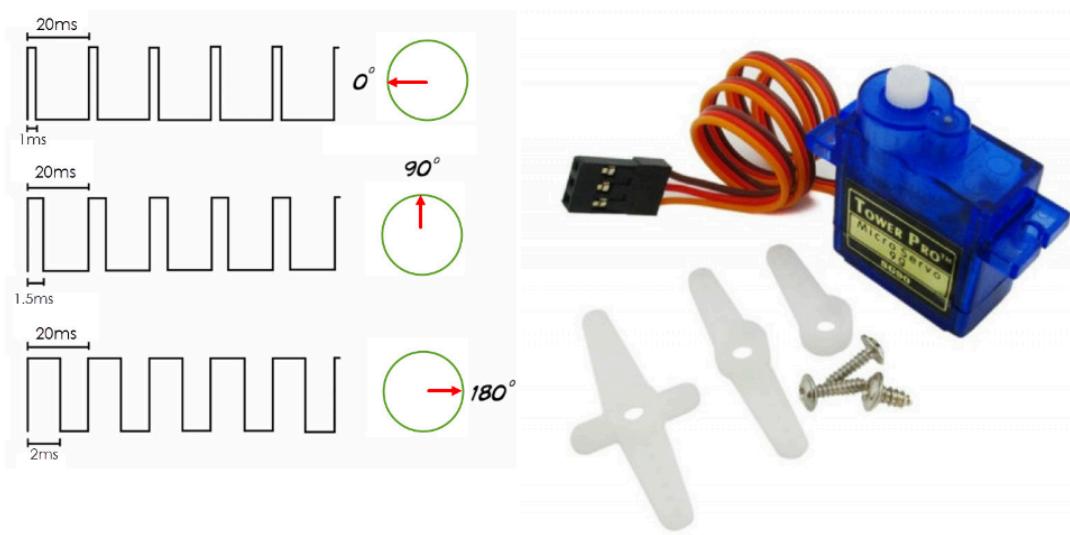


Figura 3: Controle de um servomotor por PWM.

Um servomotor é controlado por PWM, de forma a largura de pulso ajusta o ângulo do eixo. A biblioteca **Servo.h** contém as funções necessárias para configuração. Devemos criar um objeto do tipo Servo e atribuir ele em uma porta por meio do método **attach** no **setup**. No loop, o ângulo do servo é atribuído pelo método **write**, que recebe um ângulo em graus de **0 a 180**.

```

#include <Servo.h>

Servo servoMotor; // Cria um objeto da classe Servo

const int servoPin = 9; // Pino de controle do servomotor

void setup() {
    servoMotor.attach(servoPin); // Conecta o pino do servomotor ao objeto servoMotor
}

void loop() {
    // Movimento para a posição inicial
    servoMotor.write(0); // Define a posição do servo para 0 graus
    delay(1000); // Aguarde 1 segundo

    // Movimento para a posição intermediária
    servoMotor.write(90); // Define a posição do servo para 90 graus
    delay(1000); // Aguarde 1 segundo

    // Movimento para a posição final
    servoMotor.write(180); // Define a posição do servo para 180 graus
    delay(1000); // Aguarde 1 segundo
}

```

Por definição, o pulso que controlar o servo é ajustado em 50 Hz, sendo que uma largura de pulso de 544 microssegundos coloca o servo em 0 graus e um pulso de 2400 microssegundos ajusta o servo em 180 graus. Apesar desses serem valores padrões do método **attach**, é possível configurar de forma personalizada, visto que existem diferenças de servos dependendo do fabricante. Para ajustar a largura de pulso mínima e máxima, podemos usar **attach(servoPin, min, max)**.

É importante se atentar à pinagem do servomotor:

Cor do fio que sai do servo	Pino	Conecta no Arduino
Marrom	Ground	GND
Vermelho	Power	5 V
Laranja	Control	Saída Analógica (3, 5, 6, 9, 10, 11)

3. Parte prática

Considerando os circuitos da Figuras 4, elabore a prática como se pede nos itens abaixo.

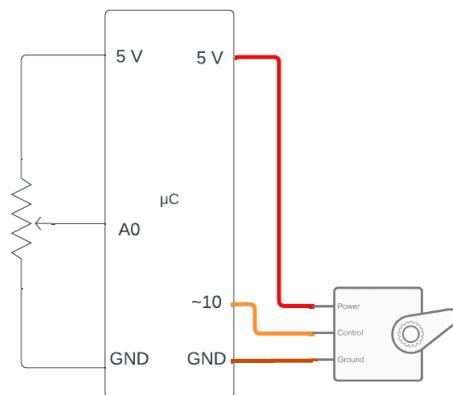


Figura 4: Circuito para a montagem prática A

- A. **Potenciômetro controlando o servomotor:** Crie um código que fará com que um potenciômetro controle o ângulo do servomotor. Por segurança faça com que o servo varie entre 15 e 165 graus. Visualize a saída analógica (PWM) no osciloscópio e a entrada do potenciômetro na serial. Qual o período mínimo e máximo da largura de pulso?
- B. **Desafio (Controle de Precisão):** Modifique o circuito anterior incluindo dois push-buttons em duas entradas digitais. Um botão irá fazer com que o servo incremente o ângulo e o outro decremente. O potenciômetro será utilizado para controlar a sensibilidade, ou seja, um parâmetro que fará com que o servo gire mais rápido ou devagar ao segurar o botão. Dica: o valor lido do potenciômetro poderá controlar o delay entre um incremento e outro. **Desafio do desafio:** inclua uma função que fará com que se ambos os botões sejam apertados ao mesmo tempo nada ocorra, mas se os botões se mantiverem pressionados por 3 segundos, o sistema entrará no modo de controle por potenciômetro (como na questão A).