



# INSTITUTO FEDERAL Paraíba

## Projeto SE-2024.2

### Equipe:

- Matheus Carneiro da Cunha
- Heitor Brunini
- Rafael Berg

## Documentação

Estas bibliotecas permitem controlar um servo motor usando PWM no ESP-IDF, ajustando a posição do servo com base na largura do pulso enviado ao seu pino de controle.

### SSR

Funções Disponíveis **servo\_hw**

1. **hw\_servo\_init( ):**
  - Configura o timer e o canal PWM.
  - Testa o servo movendo-o de 0° a 180° e de volta para 0°.
2. **hw\_servo\_set\_pulse\_width( ):**
  - Converte a largura de pulso desejada para um valor de duty cycle.
  - Define o duty cycle no canal do PWM.
3. **hw\_servo\_deinit( ):**
  - Para o PWM e pausa o timer, desligando o controle do servo.

## Funções Disponíveis **servo\_tools**

1. **servo\_init( )**:
  - Verifica se o ponteiro config é válido.
  - Chama hw\_servo\_init( ) para configurar o PWM e preparar o controle do servo.
2. **servo\_set\_angle( )**:
  - Verifica se o ponteiro config é válido e se o ângulo está entre 0° e 180°.
  - Converte o ângulo em largura de pulso (usada para gerar o PWM correto para o servo).
  - Chama hw\_servo\_set\_pulse\_width() para enviar o sinal PWM correspondente.
  - Se for bem-sucedido, atualiza config->current\_angle com o novo ângulo.
3. **servo\_get\_angle( )**:
  - Retorna o ângulo atual armazenado em config->current\_angle.
  - Útil para saber a posição atual do servo sem precisar medir fisicamente.
4. **servo\_deinit( )**:
  - Chama hw\_servo\_deinit( ) para desligar o PWM e liberar os recursos.

## Resumo

Esse código facilita o controle de um servo motor no ESP-IDF. Ele encapsula funções para inicializar, definir e obter o ângulo do servo, além de desativá-lo quando não for mais necessário. O uso do LEDC (módulo de PWM do ESP32) permite gerar o sinal de controle com precisão.