Interruptor Wi-Fi

Equipe: Beatriz Chen e Gabriel Dimant

★ Descrição

O projeto consistiu em criar um sistema com um servo-motor que pode ser acionado via Wi-Fi em um site pelo celular, que liga e desliga um interruptor de parede, a princípio conectado a uma lâmpada. Desse modo, é possível acender e desligar a luz do cômodo sem precisar se deslocar até o interruptor.

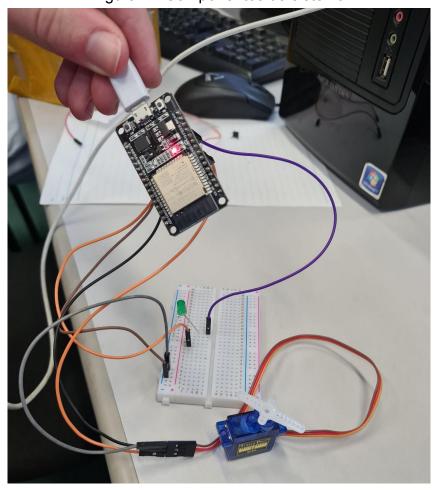


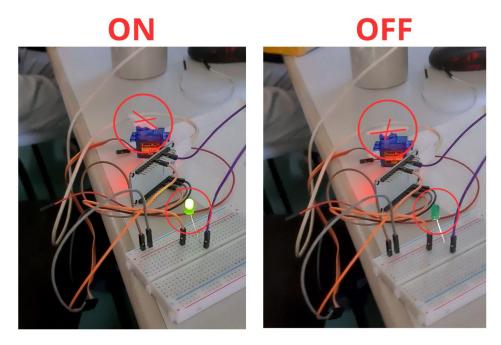
Figura 1 - Componentes do sistema

Legenda: Uso de um ESP32, um servo-motor, um LED e uma protoboard.

Figura 2 - Funcionamento do projeto

Legenda: O sistema do servo-motor e LED está conectado via Wi-Fi ao celular que controla o movimento do servo-motor e, consequentemente, o LED.

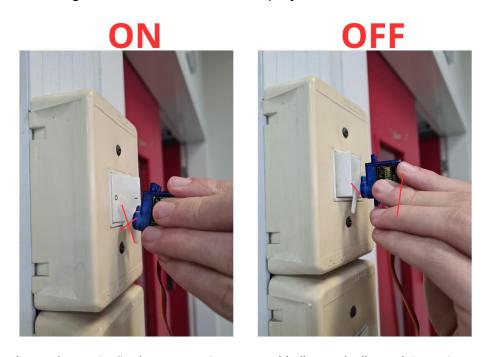
Figura 3 - Funcionamento do projeto



Legenda: O servo-motor estaria acoplado no interruptor e seu movimento acarretaria no "ligar" e "desligar" a luz do interruptor, como acontece com o LED, que liga quando se aperta o "ON" e desliga quando se aperta o "OFF".

Na vida real, o funcionamento do projeto seria mais ou menos assim:

Figura 4 - Funcionamento do projeto na realidade



Legenda: a rotação do servo-motor conseguiria ligar e desligar o interruptor.



Lista de materiais

Quantidad e	Nome	Link para referência
1	ESP32 e cabo USB	https://www.baudaeletronica.com.br/placa-doit-e sp32-bluetooth-e-wifi.html
1	Servo-motor	https://www.baudaeletronica.com.br/produto/micro-servo-9g-sg90-tower-pro.html
1	LED de qualquer cor	https://www.baudaeletronica.com.br/produto/led-difuso-3mm-verde
4	Jumpers variados	https://www.baudaeletronica.com.br/produto/65-j umpers-para-protoboard-macho-macho.html
1	Protoboard	https://www.baudaeletronica.com.br/produto/protoboard-400-pontos.html
1	Resistor 100 ohms	baudaeletronica.com.br/produto/resistor-1k-5-14 w.html

Lista de conexões

Componente	Pino da placa	
PWM do servo-motor	25	
GND do servo-motor	GND	

V+ do servo-motor 3V3

LED 22

Funcionamento dos sensores e atuadores

Servo-motor Micro Servo 9G SG90

O Micro Servo 9G SG90 é um servo-motor que opera neste projeto a partir da modulação por largura de pulso (PWM), como alternativa ao uso do conversor digital-analógico (DAC). Sua tensão de funcionamento é entre 0 a 6V e possui um torque específico para cada faixa de tensão. Possui uma rotação de 180° e velocidade de 0,11 seg/ 60 graus. Especificação técnica:

https://curtocircuito.com.br/servo-motor-9g-sg90-towerpro.html

Circuito

IN Floating IN Flo

Figura 5 - Diagrama do circuito em repouso

Figura 6 - Diagrama do circuito na posição "OFF"

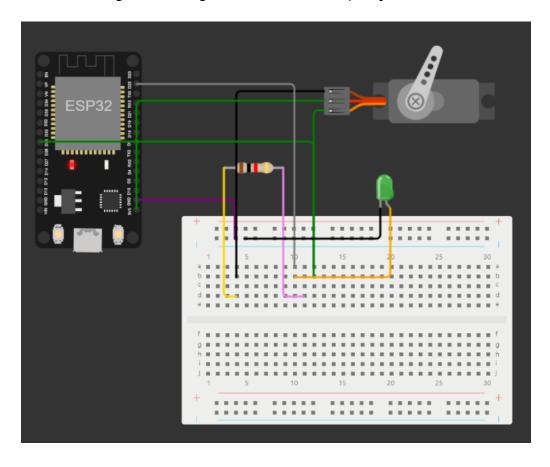
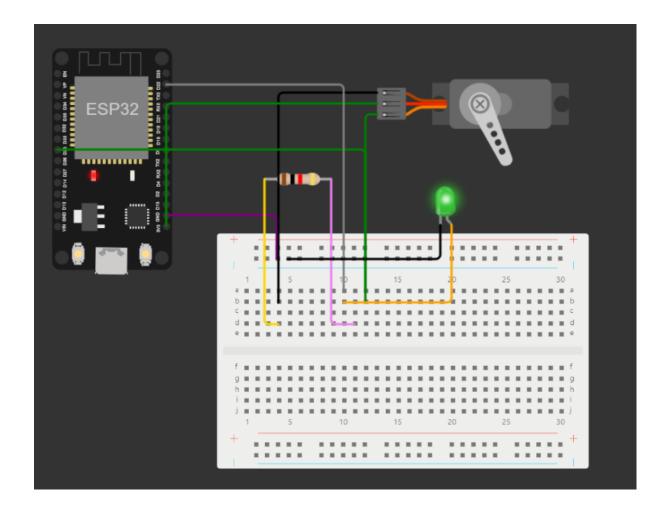


Figura 7 - Diagrama do circuito na posição "ON"



Funcionamento do sistema

Requisitos do código

Para utilizar a conexão Wi-Fi do ESP32 neste projeto, é necessário que o microcontrolador e o usuário estejam conectados na mesma rede.

Estrutura de arquivos

Neste projeto foi utilizado apenas o boot.py.

Objetivo de cada arquivo/pedaço de código

Os comentários presentes no código indicam o papel de cada pequeno bloco no programa:

```
import socket
from machine import Pin, PWM
```

```
import network
import esp
import gc
from utime import sleep
led = Pin(12, Pin.OUT)
class ESPServer:
   def init (self):
       esp.osdebug(None)
       gc.collect()
       self.station = network.WLAN(network.STA IF)
        self.station.active(True)
       self.locker state = "OFF"
   def connect(self, ssid, password):
       self.station.connect(ssid, password)
       while self.station.isconnected() == False:
       print('Connection successful')
       print(self.station.ifconfig())
   def web page(self):
name="viewport" content="width=device-width, initial-scale=1">
Helvetica; display:inline-block; margin: 0px auto; text-align: center;}
1.5rem; }.button { display: inline-block; background-color: #e7bd3b;
text-decoration: none; font-size: 30px; margin: 2px; cursor: pointer;}
<h1>Digital Locker</h1>
        Locker state: <strong>""" + self.locker state +
class="button">ON</button></a>
button2">OFF</button></a></body></html>"""
   def start(self):
```

```
p25 = Pin(25, Pin.OUT)
        motor = PWM(p25, freq=50)
        motor.duty(40)
        locker state = "OFF"
        s.bind(('', 3000))
        s.listen(2)
        while True:
            conn, addr = s.accept()
            print('Got a connection from %s' % str(addr))
            request = conn.recv(1024)
            request = str(request)
            print('Content = %s' % request)
            locker on = request.find('/?locker=on')
            locker off = request.find('/?locker=off')
                print('LOCKER ON')
                motor.duty(110)
                led.on()
            if locker off == 6:
                print('LOCKER OFF')
                motor.duty(40)
                led.off()
            response = self.web_page()
            conn.send('HTTP/1.1 200 OK\n')
            conn.send('Content-Type: text/html\n')
            conn.send('Connection: close\n\n')
            conn.sendall(response)
            conn.close()
server = ESPServer()
server.connect("", "")
server.start()
```

Contribuir no projeto

Features implementadas

Acender o LED e mexer o servo-motor ao clicar em "on" ou "off" pelo celular

Features para incrementar no projeto

Usar algum sensor que capte movimento e substitua o "clicar em 'on' e 'off'", ou seja, controlar o ligar do LED e o movimento do servo-motor através do sensor de movimento