

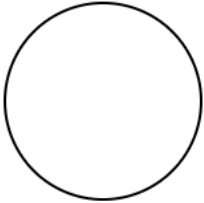


Ziel des Projekts

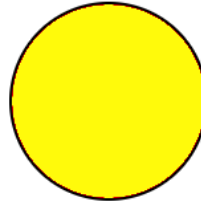
Das Programm "baut" eine Webseite, die mit der WiFi-Funktion des ESP32 angezeigt wird. Ein Klick auf den Button schaltet die LED.



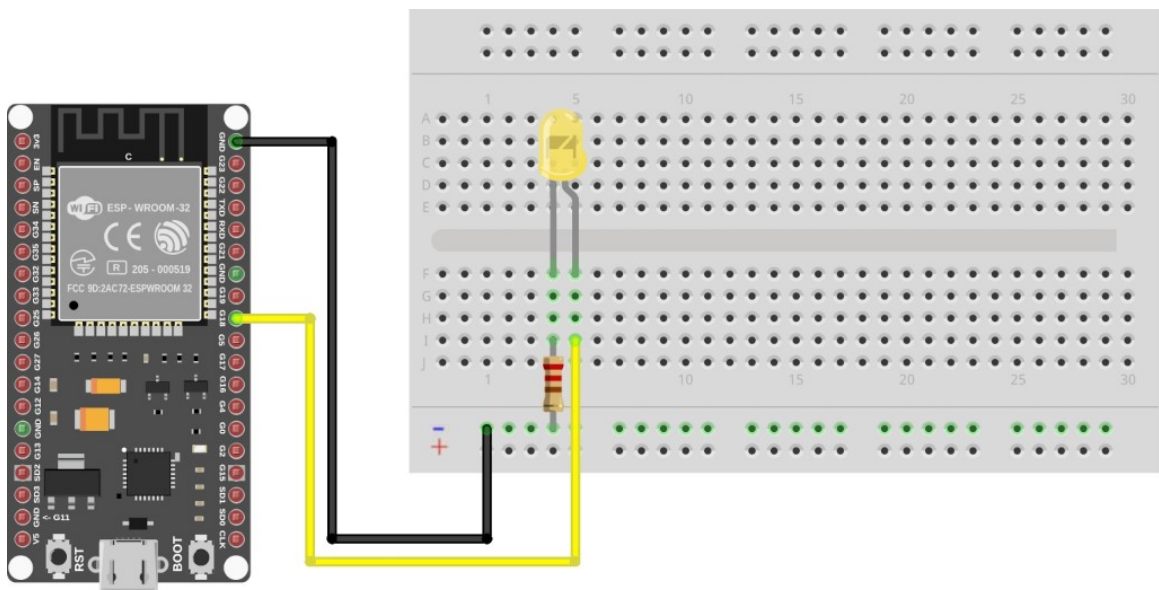
LED schalten



LED schalten



Der Schaltplan



fritzing

Das Programm

Im Programm kannst du entscheiden, ob du dem Router die Vergabe der IP-Adresse überlassen willst (statischeIP = false) oder selbst eine IP-Adresse festlegen willst (statischeIP = true).

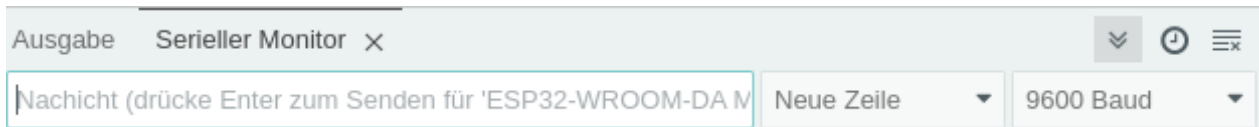
Die IP-Adresse wird im Seriellen Monitor angezeigt, du musst sie in der Adresszeile eines Browsers deiner Wahl eingeben.



...

Verbunden mit FRITZ!Box 7590 LB
IP über DHCP: 192.168.1.217

IP-Adresse mit DHCP



Verbunden mit FRITZ!Box 7590 LB
Statische IP: 192.168.1.100

Statische IP-Adresse

Der Button wird mit CSS gestaltet:

- background-color: Hintergrundfarbe
- width/height: Breite und Höhe
- cursor: Form des Cursors (pointer)
- border-radius: Form der Linie (50% -> Kreis)
- border: Breite in Pixel , Darstellung und Farbe der Linie (2px solid black)

Der Programmcode

```
#include "WiFi.h"
#include "WebServer.h"

// Router-SSID/Passwort anpassen
char Router[] = "Router_SSID";
char Passwort[] = "xxxxxxx";

WebServer Server(80);

int LED = 18;

// LED beim Start ausgeschaltet
bool LEDStatus = false;
```



```
// statischeIP = false -> IP-Adresse über DHCP vergeben
// statischeIP = true -> statische IP festlegen
bool statischeIP = false;

// ip und gateway müssen an das lokale Netz angepasst werden
IPAddress ip(192, 168, 1, 100);
IPAddress gateway(192, 168, 1, 1);
IPAddress subnet(255, 255, 255, 0);

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

  // auf serielle Verbindung warten
  while (!Serial);
  delay(1000);

  // WiFi starten
  WiFi.begin(Router, Passwort);

  // statische IP vergeben
  if (statischeIP)
  {
    WiFi.config(ip, gateway, subnet);
    Serial.print("Verbunden mit ");
    Serial.println(Router);

    // IP anzeigen
    Serial.print("Statische IP: ");
  }

  // IP über DHCP ermitteln
  else
  {
    while (WiFi.status() != WL_CONNECTED)
    {
      delay(200);
      Serial.print(".");
    }
    Serial.println();
    Serial.print("Verbunden mit ");
    Serial.println(Router);
    Serial.print("IP über DHCP: ");
  }

  // IP anzeigen
  Serial.println(WiFi.localIP());

  // Webserver starten
  // / -> Aufruf der URL, SeiteBauen -> Aufruf der Funktion
  Server.begin();
  Server.on("/", SeiteBauen);
}
```



```
// beim Aufruf der URL /LEDSchalten
// Funktion für die Schaltung der LED aufrufen
Server.on("/LEDSchalten", LEDSchalten);
}

void loop()
{
    // auf Anfragen warten
    Server.handleClient();
}

void SeiteBauen()
{
    // Seite "zusammenbauen"
    String Seite = "";
    Seite += "<h1 align=\"left\">LED schalten</h1>";

    // Link beim Anklicken des Buttons
    // Aufruf der Funktion LEDSchalten
    Seite += "<a href=\"/LEDSchalten\">";

    // LED aus -> LEDStatus false -> gelber Hintergrund
    if (LEDStatus) Seite += "<button style=\"background-color: yellow;";

    // LED an -> LEDStatus true -> weißer Hintergrund
    else Seite += "<button style=\"background-color: white;";
    Seite += " width:150px;height:150px; cursor:pointer;";
    Seite += " border-radius:50%;border: 2px solid black;\>";
    Seite += " </button>";

    // Seite anzeigen
    Server.send(200, "text/html", Seite);
}

void LEDSchalten()
{
    // LEDStatus = false -> LED einschalten
    if (!LEDStatus) digitalWrite(LED, HIGH);

    // LEDStatus = true -> LED ausschalten
    else digitalWrite(LED, LOW);
    LEDStatus = !LEDStatus;

    // Seite neu aufbauen
    SeiteBauen();
}
```