

KATOLÍCKA UNIVERZITA V RUŽOMBERKU
PEDAGOGICKÁ FAKULTA

Ovládanie RGB LED pásu pomocou ESP32

Projektová dokumentácia

Vypracovali: Maroš Balcerčík
Dávid Širaj

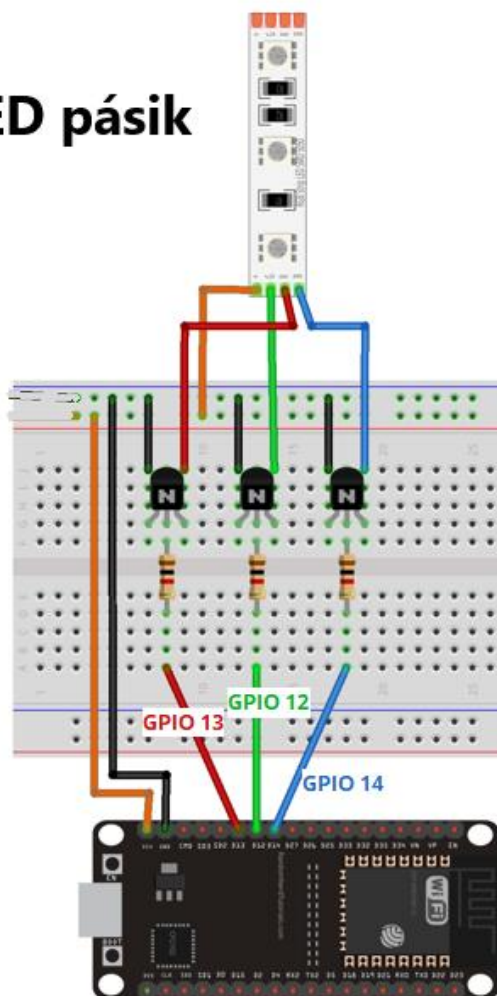
2024

Ciel':

Ciel'om nášho projektu, bolo vytvorenie systému zabezpečujúce ovládanie RGB LED pásika. Projekt bol uskutočnený v rozhraní Arduino IDE s použitím ESP-32-WROOM-32D vývojovej dosky s Wifi a bluetooth, ktorý host'uje web server na ktorom sa nám zobrazí stránka s možnosťou zmeny farieb pre náš RGB LED Pásik.

Schéma zapojenia:

5V LED pásik



Súčiastky:

1. Breadboard – prepojovacie pole
2. ESP-32-WROOM-32D
3. RGB LED pásik (5V)
4. Prepájacie drôty
5. Micro USB kábel
6. 3x NPN tranzistor
7. 3x 1k ohm rezistor

Postup práce:

Ďalším krokom bolo naprogramovať ESP32 v Arduino IDE. Následne sme premenili náš 2D náčrt na 3D model. Vytvorili sme tak spojenie medzi jednotlivými potrebnými súčiastkami. To znamenalo jednotlivé kroky:

1. Z ESP 32 sme vyviedli napätie 5V a GND (zem/uzemnenie)
2. Na fotkách, ktoré sú na vo fotodokumentácii je vidieť množstvo káblov, kde červený je použitý pre napájanie a čierny je použitý pre GND.
3. Následne prišlo utrpenie v podobe zapojenia LED pásiku, tranzistorov a rezistorov.
4. Pre napájanie LED pásiku, je opäť použitý červený káblík. Pre jednotlivé farby boli použité tieto káblíky: biely pre zelenú, čierny pre červenú a hnedý pre modrú.
5. Tieto káblíky použijeme pre napojenie na tranzistory a cez výstupy týchto tranzistorov a rezistorov, priradíme jednotlivé GPIO porty na ESP 32.
6. Z rezistoru, ktorý určuje červenú farbu je cez žltý káblík priradený GPIO 13, z rezistoru pre zelenú je cez fialový káblík priradený GPIO 12 a rezistoru pre modrú je cez zelený káblík priradený GPIO 14.
7. Po úspešnom zapojení sme prešli k písaniu kódu, ktorý je viditeľný v nasledujúcej časti.

<https://randomnerdtutorials.com/esp32-esp8266-rgb-led-strip-web-server/>

Napísanie kódu:

```
#include <WiFi.h>
```

```
const char* ssid    = "Nahradiť vašou wifi názvom";
```

```
const char* password = "Zadajte vaše heslo";
```

```
WiFiServer server(80);
```

```
String redString = "0";  
String greenString = "0";  
String blueString = "0";  
int pos1 = 0;  
int pos2 = 0;  
int pos3 = 0;  
int pos4 = 0;
```

```
String header;
```

```
const int redPin = 13;  
const int greenPin = 12;  
const int bluePin = 14;
```

```
const int freq = 5000;  
const int redChannel = 0;  
const int greenChannel = 1;  
const int blueChannel = 2;
```

```
const int resolution = 8;
```

```
unsigned long currentTime = millis();
```

```

unsigned long previousTime = 0;

const long timeoutTime = 2000;

void setup() {
  Serial.begin(115200);

  ledcSetup(redChannel, freq, resolution);
  ledcSetup(greenChannel, freq, resolution);
  ledcSetup(blueChannel, freq, resolution);

  ledcAttachPin(redPin, redChannel);
  ledcAttachPin(greenPin, greenChannel);
  ledcAttachPin(bluePin, blueChannel);

  Serial.print("Pripájam na ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("WiFi pripojené.");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());

```

```

server.begin();
}

void loop(){
  WiFiClient client = server.available();

  if (client) {
    currentTime = millis();
    previousTime = currentTime;
    Serial.println("Nový uživatel.");
    String currentLine = "";
    while (client.connected() && currentTime - previousTime <= timeoutTime) {
      currentTime = millis();
      if (client.available()) {
        char c = client.read();
        Serial.write(c);
        header += c;
        if (c == '\n') {

          if (currentLine.length() == 0) {

            client.println("HTTP/1.1 200 OK");
            client.println("Content-type:text/html");
            client.println("Connection: close");
            client.println();

            client.println("<!DOCTYPE html><html>");

```

```

        client.println("<head><meta name=\"viewport\" content=\"width=device-width,
initial-scale=1\">");

        client.println("<link rel=\"icon\" href=\"data:;\">");

        client.println("<link rel=\"stylesheet\"
href=\"https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css\">");

        client.println("<script
src=\"https://cdnjs.cloudflare.com/ajax/libs/jscolor/2.0.4/jscolor.min.js\"></script>");

        client.println("</head><body><div class=\"container\"><div
class=\"row\"><h1>ESP LED Pasik</h1></div>");

        client.println("<a class=\"btn btn-primary btn-lg\" href=\"\"#\"
id=\"change_color\" role=\"button\">Zmena farby</a> ");

        client.println("<input class=\"jscolor {onFineChange:'update(this)'}\"
id=\"rgb\"></div>");

        client.println("<script>function update(picker)
{document.getElementById('rgb').innerHTML = Math.round(picker.rgb[0]) + ', ' +
Math.round(picker.rgb[1]) + ', ' + Math.round(picker.rgb[2]);}");

        client.println("document.getElementById(\"change_color\").href=\"?r\" +
Math.round(picker.rgb[0]) + \"g\" + Math.round(picker.rgb[1]) + \"b\" +
Math.round(picker.rgb[2]) + \"&\";}</script></body></html>");

        client.println();

```

```

if(header.indexOf("GET /?r") >= 0) {

    pos1 = header.indexOf('r');

    pos2 = header.indexOf('g');

    pos3 = header.indexOf('b');

    pos4 = header.indexOf('&');

    redString = header.substring(pos1+1, pos2);

    greenString = header.substring(pos2+1, pos3);

    blueString = header.substring(pos3+1, pos4);

    ledcWrite(redChannel, redString.toInt());

```

```

        ledcWrite(greenChannel, greenString.toInt());
        ledcWrite(blueChannel, blueString.toInt());
    }

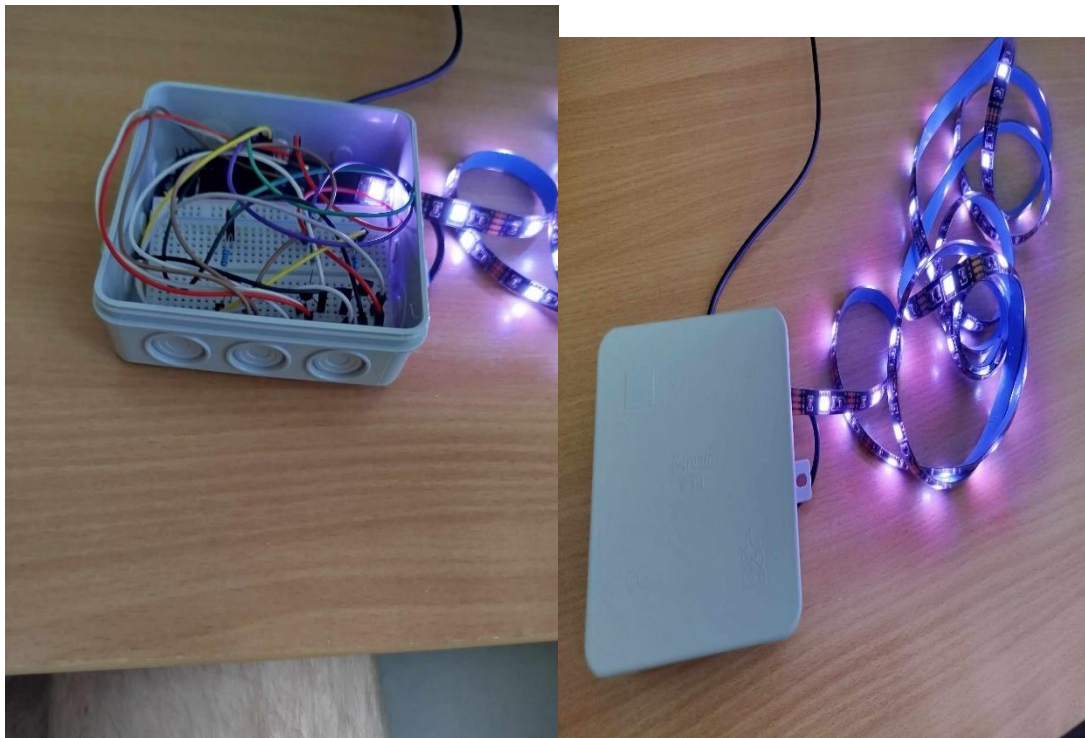
    break;
} else {
    currentLine = "";
}
} else if (c != '\r') {
    currentLine += c;
}
}
}

header = "";

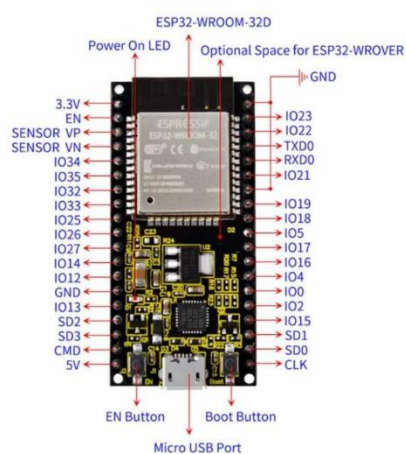
client.stop();
Serial.println("Užívateľ sa odpojil.");
Serial.println("");
}
}

```


Fotodokumentácia:



Použitý datasheet:



Link na GitHub: <https://github.com/DavidSiraj/Sibali>

Zdroje:

<https://randomnerdtutorials.com/esp32-esp8266-rgb-led-strip-web-server/>