

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
Fakulta informačních technologií



**IMP - Mikroprocesorové a vstavané systémy**  
**M09 - Ovládanie RGB Led**

2023/2024

# Obsah

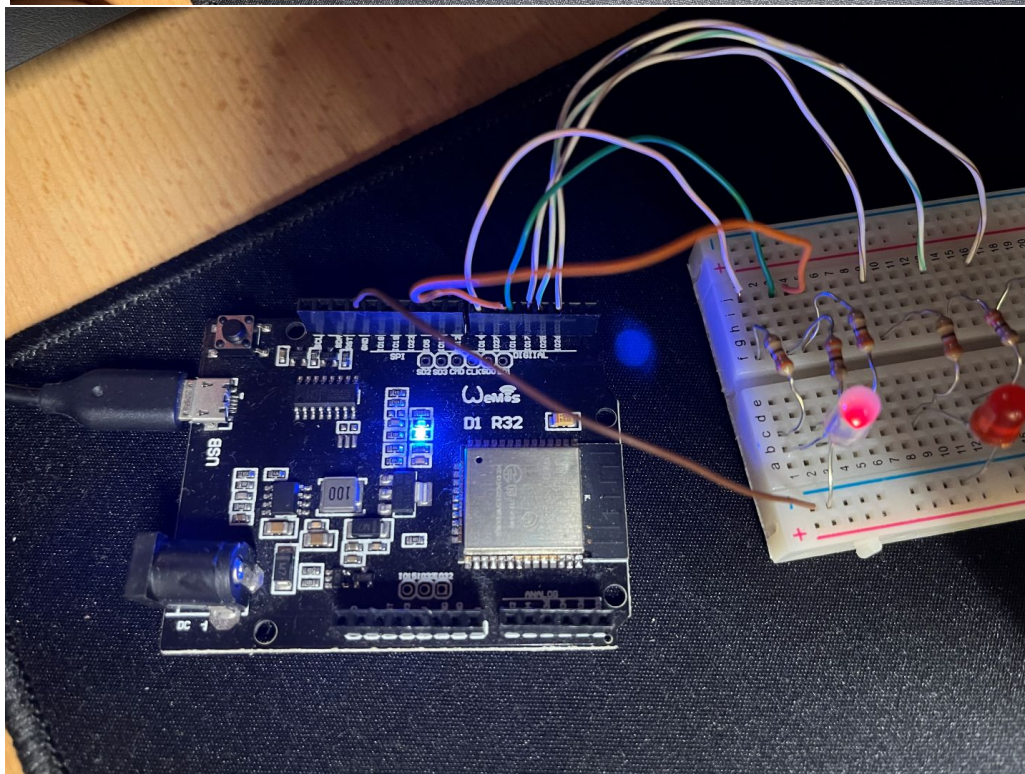
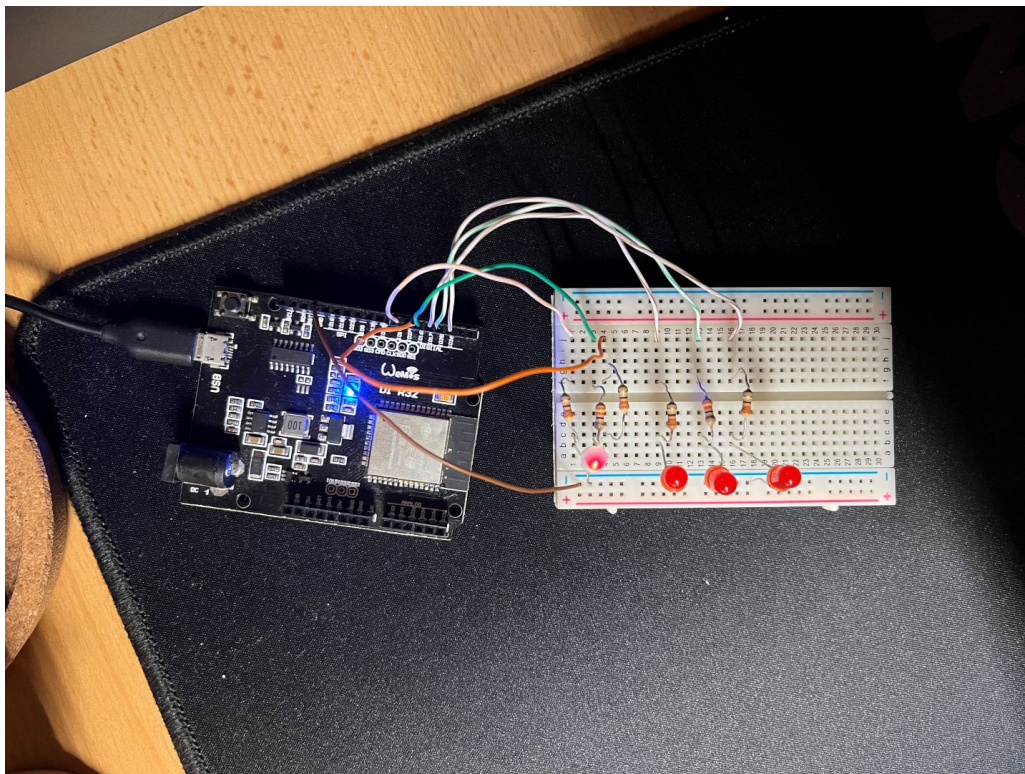
<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Návod</b>	<b>3</b>
2.1	Zapojenie . . . . .	3
2.2	Spustenie . . . . .	4
2.3	Ovládanie . . . . .	4
2.3.1	Režimy LED a RGB diód . . . . .	4
2.3.2	Rýchlosti . . . . .	4
2.3.3	Ovládač . . . . .	5
2.3.4	Pripojenie k zariadeniu . . . . .	5
2.3.5	Ovládanie LED a RGB diód . . . . .	5
2.3.6	Odpojenie od zariadenia . . . . .	5
2.4	Videoprezentácia funkčnosti . . . . .	6
<b>3</b>	<b>Implementácia</b>	<b>7</b>
3.1	main.cpp . . . . .	7
3.1.1	class MyCharacteristicCallbacks . . . . .	7
3.1.2	class MyServerCallbacks . . . . .	8
3.1.3	void setup() . . . . .	8
3.1.4	void loop() . . . . .	8
3.1.5	void setRGB(int red, int green, int blue) . . . . .	8
3.1.6	void RGBFire() . . . . .	8
3.1.7	void RGBFlicker() . . . . .	8
3.1.8	void RGBBreathe() . . . . .	9
3.1.9	void RGBPolice() . . . . .	9
3.1.10	void RGBFade() . . . . .	9
3.1.11	void LEDRaceStart() . . . . .	9
3.1.12	void LEDBinary() . . . . .	9
3.1.13	void LEDChase() . . . . .	9
3.1.14	void LEDRise() . . . . .	9
3.2	index.html . . . . .	10
	<b>Literatúra</b>	<b>11</b>

## Úvod

Tento projekt vznikol ako výsledok zadania z predmetu Mikroprocesorové a vstavané systémy (IMP)[3] na Fakulte informačných technológií VUT v Brne[2]. Úlohou tohto projektu bolo vytvoriť aplikáciu na autonómne ovládanie skupiny LED diód a jednej RGB LED cez Bluetooth rozhranie s použitím dosky ESP32. Aplikácia mala umožniť spúšťať rôzne animácie s nastaviteľnou rýchlosťou. Ovládanie malo byť realizované prostredníctvom jednoduchšej webovej stránky optimalizovanej pre mobilné zariadenia.

# Návod

## Zapojenie



## Spustenie

K spusteniu budete potrebovať mikrokontrolér ESP32 (testované na Wemos D1 R32[4]). Ďalej budete potrebovať zariadenie podporujúce Bluetooth. Na spustenie ovládača, kvôli použitiu API Web bluetooth[5], budete potrebovať taktiež jeden z nasledujúcich prehliadačov:

- Google Chrome
- Microsoft Edge
- Opera
- Samsung Internet
- UC Browser pre Android
- Baidu Browser

(Ovládač je optimalizovaný pre mobilné zariadenia a bol testovaný hlavne na prehliadači Google Chrome.)

## Ovládanie

### Režimy LED a RGB diód

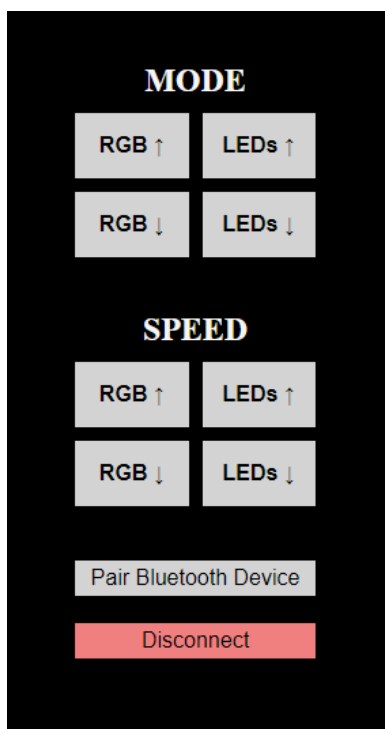
LED a RGB diódy majú nasledovné režimy:

- RGB Režimy:
  - Imitácia ohňa
  - Blikanie
  - Dýchanie
  - Policačné svetlá
  - Prechod RGB spektrom
- LED Režimy:
  - Štart pretekov
  - Postupné zapínanie a vypínanie
  - Binárny režim
  - Postupné rozosvecovanie

### Rýchlosti

Existujú celkom 4 rýchlosti pre LED a RGB režimy. Rýchlosti medzi LED a RGB sú nezávislé. Pri prvotnom spustení je rýchlosť nastavená na úroveň 2.

## Ovládač



### Pripojenie k zariadeniu

Pre pripojenie k LED ovládaču postupujte podľa nasledujúcich krokov:

1. Otvorte webový prehliadač (jeden zo zoznamu podporovaných prehliadačov) a otvorte v ňom súbor index.html.
2. Kliknite na tlačidlo **Pair Bluetooth Device** na stránke.
3. Vyberte svoje zariadenie v dialógovom okne Bluetooth a pripojte sa k nemu.

### Ovládanie LED a RGB diód

Po úspešnom pripojení môžete ovládať LED a RGB diódy nasledovne:

- Pre zmenu režimu LED alebo RGB diód použite tlačidlá **LED Mode Up/Down** alebo **RGB Mode Up/Down**.
- Na zmenu rýchlosti svietenia použite tlačidlá **LED Speed Up/Down** alebo **RGB Speed Up/Down**.

### Odpojenie od zariadenia

Ak chcete odpojiť ovládač od Bluetooth zariadenia, stlačte tlačidlo **Disconnect**.

## Videoprezentácia funkčnosti

Link na youtube video znázorňujúce zapojenie obvodu spoločne s prezentáciou funkčnosti: <https://youtu.be/9Q2A-jNEQss?si=hiww69qztt4MYszU>

## Implementácia

K implementácii bolo použité vývojové rozhranie Visual Studio Code rozšírené o PlatformIO IDE[6]. Programovanie prebiehalo na platforme Arduino[1]. Implementácia sa skladá z 2 častí:

- main.cpp
- index.html

### main.cpp

Tento program sa skladá z niekoľko menších častí kódu:

- class MyCharacteristicCallbacks
- class MyServerCallbacks
- void setup()
- void loop()
- void setRGB(int red, int green, int blue)
- void RGBFire()
- void RGBFlicker()
- void RGBBreathe()
- void RGBPolice()
- void RGBFade()
- void LEDRaceStart()
- void LEDBinary()
- void LEDChase()
- void LEDRise()

### class MyCharacteristicCallbacks

Trieda `MyCharacteristicCallbacks`, ktorá je odvodená od `BLECharacteristicCallbacks`, je navrhnutá na spracovanie prichádzajúcich dát cez Bluetooth Low Energy (BLE) charakteristiky, s osobitným zameraním na ovládanie LED a RGB diód. Funkcia `onWrite` v tejto triede sa aktivuje, keď sú na charakteristiku zapisované dáta. Táto funkcia potom číta prijaté dáta a vykonáva príslušné akcie podľa prijatých príkazov.

Funkcia rozlišuje rôzne príkazy:

**LED SPEED UP** a **LED SPEED DOWN**: Tieto príkazy zvyšujú alebo znižujú interval blikania LED diód. Pre každý príkaz sa podľa aktuálnej hodnoty `LEDInterval` nastaví nová hodnota v preddefinovaných krokoch (200, 400, 600, 800 milisekúnd).

**RGB SPEED UP** a **RGB SPEED DOWN**: Podobne ako u LED, tieto príkazy zvyšujú alebo znižujú interval zmeny farby RGB diódy. Interval sa nastavuje v krokoch (30, 60, 90, 120 milisekúnd).

**LED MODE UP** a **LED MODE DOWN**: Tieto príkazy menia efekt zobrazovania na LED diódach. Funkcia cyklicky prechádza cez rôzne režimy (dokopy 4).



**RGB MODE UP** a **RGB MODE DOWN**: Analogicky k LED diódam, tieto príkazy menia režim zobrazovania RGB diódy, pričom prechádzajú cez rôzne efekty (dokopy 5).

Každý príkaz sa porovnáva s prijatými dátami a na základe zhody sa vykoná príslušná akcia. Táto funkcia umožňuje flexibilné ovládanie chovania LED a RGB diód prostredníctvom Bluetooth komunikácie.

### **class MyServerCallbacks**

Trieda **MyServerCallbacks** je odvodená od základnej triedy **BLEServerCallbacks**, ktorá poskytuje spätné volania pre udalosti Bluetooth Low Energy (BLE) servera. V tejto triede je implementovaná funkcia **onDisconnect**, ktorá sa aktivuje v prípade, keď dôjde k odpojeniu klienta od BLE servera.

V podstate táto funkcia zabezpečuje, že BLE server zostáva viditeľný a dostupný pre ďalšie zariadenia aj po odpojení existujúceho klienta.

### **void setup()**

Funkcia **setup()** je určená pre inicializáciu a nastavenie Bluetooth Low Energy (BLE) servera a súvisiacich komponentov na mikrokontroléri ESP32.

### **void loop()**

Funkcia **loop()** je hlavná slučka mikrokontroléra, ktorá neustále aktualizuje stavy LED a RGB diód. Táto funkcia neustále kontroluje čas a na základe uplynutého intervalu aktualizuje efekty na LED a RGB diódach. Časový interval aktualizácie vie užívateľ upraviť pomocou [index.html](#).

### **void setRGB(int red, int green, int blue)**

Funkcia **setRGB(int red, int green, int blue)** je určená na nastavenie farby RGB LED diódy. Vstupnými parametrami sú hodnoty pre červenú (red), zelenú (green) a modrú (blue) farbu. Tieto hodnoty sú potom použité v funkcii **analogWrite** pre nastavenie intenzity svetla jednotlivých pinov, ktoré ovládajú červenú (RED PIN), zelenú (GREEN PIN) a modrú (BLUE PIN) komponentu RGB LED. Výsledkom je zmena farby RGB LED diódy podľa zadaných hodnôt.

### **void RGBFire()**

Funkcia **RGBFire()** simuluje efekt plameňa na RGB LED dióde. Používa časovač na kontrolu, či je čas zmeniť "blikanie" plameňa, pričom interval blikania je náhodne generovaný v rámci nastaveného intervalu **RGBinterval**. Jas plameňa sa simuluje náhodným nastavením intenzity svetla, pričom červená zložka je vždy na maxime (255), zelená sa mení v závislosti od intenzity jasu a modrá zložka zostáva vypnutá. Na záver funkcia nastaví farbu RGB LED na aktuálne hodnoty, čím vytvára efekt blikajúceho plameňa.

### **void RGBFlicker()**

Funkcia **RGBFlicker()** vytvára efekt blikania pre RGB LED diódu. Periodicky táto funkcia náhodne mení farbu RGB LED diódy generovaním náhodných hodnôt pre červenú (**flickerRed**), zelenú (**flickerGreen**) a modrú (**flickerBlue**) komponentu. Tieto náhodné hodnoty sú potom aplikované na RGB LED pomocou funkcie **setRGB()**, čím sa dosiahne efekt blikania s meniacimi sa farbami.

### **void RGBBreathe()**

Funkcia `RGBBreathe()` riadi efekt "dýchania" pre RGB LED diódu. Funkcia postupne mení farbu LED od náhodne vygenerovanej farby po úplné stmavenie a späť, čím vytvára efekt dýchania. Intenzita svetla sa mení podľa prednastavených intervalov pre zosvetlenie, udržanie a zhasnutie. Po dokončení cyklu sa generujú nové náhodné hodnoty farieb a celý proces sa opakuje. Funkcia využíva aktuálny čas a postupné zvyšovanie alebo znižovanie intenzity každej farby na vytvorenie plynulého prechodu.

### **void RGBPolice()**

Funkcia `RGBPolice()` simuluje blikajúce policajné svetlá. Využíva časovač založený na `millis()` na kontrolu, či uplynul definovaný interval (`RGBinterval`). Blikanie prebieha v osmičkovom cykle, kde sa striedajú červená a modrá farba, pričom každá farba bliká dvakrát po sebe, a potom sa LED vypne na jednu fázu. Na konci cyklu sa premenná `blink` resetuje, čím sa zabezpečuje neustále opakovanie blikania.

### **void RGBFade()**

Funkcia `RGBFade()` je určená na postupné menenie farieb RGB LED diódy. Používa časové intervaly (definované premennou `fadeInterval`) na zmenu farebných hodnôt, pričom postupne prechádza medzi kombináciami červenej, zelenej a modrej. V každom cykle sa farebné hodnoty menia tak, aby postupne prešli od zelenej cez červenú a modrú späť na zelenú. Intenzita jednotlivých farieb sa mení v krokoch podľa premennej `colorIncrement`, čím sa dosiahne efekt plynulého prechodu medzi farbami.

### **void LEDRaceStart()**

Funkcia `LEDRaceStart()` simuluje efekt štartu pretekov pomocou troch LED diód. Vo funkcii sa používa premenná `raceState`, ktorá riadi postupné zapínanie jednotlivých LED diód (`LED PIN1`, `LED PIN2`, `LED PIN3`) v rôznych stavoch. Každý prípad v príkaze `switch` zapína alebo vypína LED diódy v určenom poradí, čím vytvára efekt postupného štartu. Po prejdení všetkých stavov sa `raceState` resetuje na nulu, čo umožňuje opakovanie efektu.

### **void LEDBinary()**

Funkcia `LEDBinary()` slúži na zobrazovanie čísel v binárnom formáte pomocou troch LED diód. V tejto funkcii sa používa premenná `number`, ktorá postupne prechádza hodnotami od 0 do 7 (binárne 000 až 111). Každý bit čísla `number` sa mapuje na jednu z troch LED diód. Pomocou operácie `analogWrite` sa nastavuje intenzita svetla pre každú LED diódu - zapnutá (255) alebo vypnutá (0) - v závislosti od toho, či príslušný bit v čísle `number` je 1 alebo 0.

### **void LEDChase()**

Funkcia `LEDChase()` vytvára efekt prenasledovania na troch LED diódach. Využíva statické premenné `chaseState` a `increment` na kontrolu smeru a postupu efektu. V závislosti od hodnoty `chaseState`, funkcia zapína jednu z LED diód na plný výkon (255), zatiaľ čo ostatné dve sú vypnuté (0). Po dosiahnutí konca sekvencie (pri hodnote `chaseState == 2`) sa zmení smer (`increment` sa nastaví na `false`), a efekt sa opakuje v opačnom poradí.

### **void LEDRise()**

Funkcia `LEDRise()` slúži na postupné rozsvietenie troch LED diód s rôznou intenzitou svetla. Používa premenné `brightness` na sledovanie intenzity svetla a `previousMillis` na meranie časového intervalu. S každým volaním funkcie sa zvyšuje hodnota `brightness` o fixný krok (v našom prípade 23) a v

závislosti od jej hodnoty sa postupne zvyšuje intenzita svetla na jednotlivých LED. Keď **brightness** dosiahne určitú hranicu, intenzita svetla sa resetuje a postup začína odznova.

## **index.html**

Táto HTML stránka slúži ako rozhranie pre ovládanie LED a RGB diód cez Bluetooth. Na stránke sú tlačidlá, ktoré umožňujú užívateľovi meniť režim a rýchlosť LED a RGB diód, ako aj pripojiť a odpojiť Bluetooth zariadenie. JavaScriptový kód obsahuje funkcie na pripojenie k Bluetooth zariadeniu, odoslanie konkrétnych príkazov a odpojenie od zariadenia. Poslucháče udalostí (**event listeners**) sú priradené k jednotlivým tlačidlám, aby reagovali na kliknutia užívateľa a vykonávali príslušné akcie, ako napríklad odoslanie príkazu **RGB SPEED UP** alebo **LED MODE DOWN** k ESP32 zariadeniu. Toto rozhranie poskytuje jednoduchý a intuitívny spôsob, ako ovládať funkcie LED a RGB diód na diaľku prostredníctvom Bluetooth pripojenia z vášho osobného zariadenia.

## Literatúra

- [1] Arduino®. <https://www.arduino.cc/>, 2023. [Online].
- [2] Fakulta informačních technologií, VUT v Brně. <http://www.fit.vutbr.cz/.cs>, 2023. [Online].
- [3] Mikroprocesorové a vestavěné systémy, FIT VUT v Brně. <https://www.fit.vut.cz/study/course/IMP/.cs>, 2023. [Online; zobrazené 13.12.2023].
- [4] Wemos d1 r32, PlatformIO.org. [https://docs.platformio.org/en/latest/boards/espressif32/wemos\\_d1\\_uno32.html](https://docs.platformio.org/en/latest/boards/espressif32/wemos_d1_uno32.html), 2023. [Online].
- [5] François Beaufort. Web bluetooth api, Communicating with Bluetooth devices over JavaScript. <https://developer.chrome.com/docs/capabilities/bluetooth>, 2015. [Online; zobrazené 13.12.2023].
- [6] platformio.org. Platformio-ide. <https://marketplace.visualstudio.com/items?itemName=platformio.platformio-ide>, 2023. [Online, naposledy aktualizované 2023-12-11, 23:17:11].