|  |  |  |
| --- | --- | --- |
|  | | |
| Závěrečná studijní práce  dokumentace | | |
| Ovládání LED pásků použitím WiFi | | |
| Martin Holuša | | |
|  | | |
|  | |  |
| Obor: | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| Třída:  Školní rok: | IT4  2016/2017 | |

#### Poděkování

*Chtěl bych poděkovat všem učitelům, kteří se podíleli na pomoci při mém závěrečném projektu. Především pak panu Ing. Petru Grussmannovi za pevné nervy a návrhu na zlepšení. Panu Mgr. Marcelu Godovskému za zprostředkování součásteka na závěr bych chtěl poděkovat svým spolužákům, kteří mi pomohli při řešení některých chyb mého projektu.*

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité   
informační zdroje.

Souhlasím, aby tato studijní práce byla použita kvýukovým účelům na Střední průmyslové   
a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2016

*podpis autora práce*

ANOTACE

Cílem mého projektu bylo napsat aplikaci, která by umožnila ovládání RGB LED pásků použitím WiFi. Aplikace byla napsaná v jazyce Arduino. Přes aplikaci můžete nastavit jakoukoliv barvu na LED páskách. LED pásky jsou řízeny pomocí čipu ESP8266. LED pásky můžete ovládat přes mobil, tablet, notebook a tak dále. Stačí se připojit na WiFi, které vytvoří čip ESP.

OBSAH

Obsah

[Úvod 6](#_Toc471749131)

[1 Výroba kontroléru na led pásky 7](#_Toc471749132)

[2 Využité technologie 8](#_Toc471749133)

[2.1 VYBAVENÍ 8](#_Toc471749134)

[2.1.1 PROCESOR 8](#_Toc471749135)

[2.1.2 LED PÁSKY 9](#_Toc471749136)

[2.1.3 TRANZISTOR 10](#_Toc471749137)

[2.2 Software 10](#_Toc471749138)

[2.3 Fritzing 11](#_Toc471749139)

[3 ZPŮSOBY ŘEŠENÍ A POUŽITÉ POSTUPY 12](#_Toc471749141)

[3.1 SEZNAM NUTNÝCH SOUČÁSTEK 12](#_Toc471749142)

[3.2 SCHÉMA ZAPOJENÍ 12](#_Toc471749143)

[3.3 Ukázka kódu 13](#_Toc471749144)

[3.1 Webový server a webový klient 13](#_Toc471749145)

[3.2 Webová aplikace 14](#_Toc471749146)

[3.3 Ukázka HTML 14](#_Toc471749147)

[3.4 Napájení 14](#_Toc471749148)

[Závěr 15](#_Toc471749149)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 16](#_Toc471749150)

[Seznam příloh 17](#_Toc471749151)

# Úvod

Většina pásků již mají svůj kontrolér, který má dálkový ovladač. Nicméně aby byla připojena WiFi budeme dělat nový kontrolér. K ovládání těchto pásků potřebujeme mikrokontrolér, některé tranzistory nebo MOSFET a WiFi.

Mohli bychom použít Arduino desku, ale to se může prodražit, zvláště poté, co přidáte do nákladů na Ethernet nebo WiFi štít. Můžeme to udělat lépe a použít relativně nový mikroprocesor ESP8266. Tato deska je úžasná. Je velmi levná a má WiFi se snadno použitelnou knihovnou. Můžeme ho naprogramovat stejně jako Arduino s Arduino IDE. Také již má spoustu digitálních vstupních / výstupních pinů.

První kapitolu své dokumentace věnuji postupu vytvoření kontroléru na ovládání RGB LED pásku, poté použitým technologiím, následně popisuji svůj postup při řešení problému.

# Výroba kontroléru na led pásky

Cílem tohoto projektu bylo vytvoření plošného spoje, naprogramování mikrokontroléru a následné vytvoření softwarové aplikace pro ovládání RGB LED pásku.

Na začátku projektu bylo velice důležité vybrat si správné technologie, které budou použity. První věc, kterou jsem musel udělat, bylo představit si, jak celé zařízení bude pracovat, jak to bude vypadat ve skutečnosti. Po utvoření představy bylo nutné vybrat si správný mikroprocesor, který celé zařízení bude řídit. Po uvážení jsem si zvolil procesor ESP8266 NodeMCU V3.

Dalším krokem bylo nakreslení plošného spoje se všemi potřebnými součástkami. Při kreslení spoje jsem několikrát měnil finální podobu.

Po nakreslení schématu zapojení celého zařízení bylo načase nakoupení všech potřebných součástek. Nejdříve jsem zapojení všech součástek zkoušel na nepájivém kontaktním poli, kde jsem si otestoval funkčnost zapojení.

Poté nastal správný čas na psaní programu. Při volbě vhodného programovacího jazyka, jsem se snažil vybrat jazyk, který by byl vhodný pro práci s mikrokontroléry a zároveň by byl vyhovoval i mým požadavkům (schopnostem). Po uvážení všech požadavků jsem se rozhodl pro jazyk Wiring, používaný si zejména mikrokontroléry Arduino, ale je využitelný i při programování mikrokontrolérů ESP8266.Při následné volbě vývojového prostředí padla má volba na program Arduino IDE. Pár projektů jsem zde již zkoušel, takže to byla pro mě nejvhodnější volba.

Po ověření funkčnosti na kontaktním poli, jsem se rozhodl pro vytvoření plošného spoje. Po zasazení součástek do těchto otvorů už jen stačí propojit správně součástky podle schématu zapojení cínem, případně jiným vodičem určeným pro pájení.

Po dokončení všech částí už jen zbývalo ho vyzkoušet v praxi. Zařízení jsem dal do krabičky, vyvrtal otvor pro napájecí kabel (a vyzkoušel doma při dlouhodobém fungování).

# Využité technologie

## VYBAVENÍ

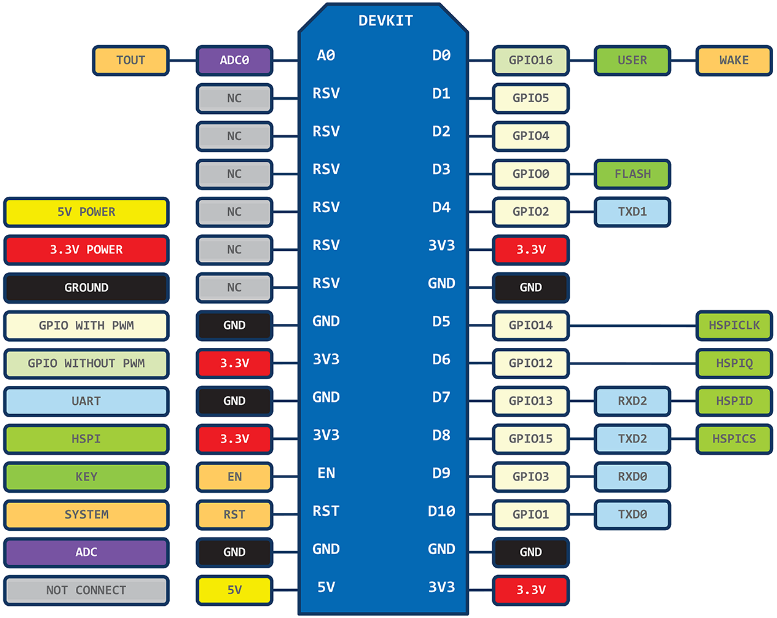
### PROCESOR

ESP8266 NodeMCU V3 je jedním z nejlevnějších modulů s připojením na WiFi na trhu. V3 neboli Version3 je nejnovější verze tohoto modulu. Modul je vytvořen na čipu ESP12E.

Piny: 15-pinů s přístupem k GPIO, SPI, UART, ADC, a napájecích kolíků.

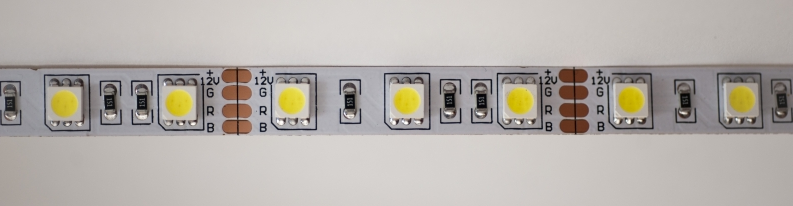
Parametry:

* 4 MB FLASH paměť
* Komunikační napětí 3,3 V
* Wireless 802.11 b/g/n



Obr.2 NodeMCU v3 piny

### LED PÁSKY

Použité LED pásky jsou typu SMD 5050. Pásek je vyroben z pružného materiálu. LED pásek můžeme zkrátit na předem určených úsecích, takže si je naměříme podle sebe. Z LED pásku jsou vyvedeny čtyři dráty: 12 V, R, G a B. LED pásky ovládáme tím, že dodáme 12 V na napájecí pin a pak uzemníme kterýkoli jiný pin na dokončení obvodu, což způsobuje, že se rozsvítí odpovídající barva. Například, pokud jsme uzemnili R pin pak se rozsvítí plnou červenou.

Obr. 3 SMD 5050 LED diody



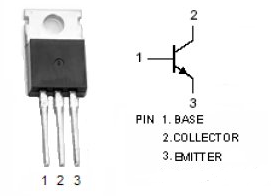
Obr. 4 LED pásek

### TRANZISTOR

Tranzistor je polovodičová součástka, kterou tvoří dvojice přechodů PN. Tranzistory jsou základní aktivní součástky, které se používají jako zesilovače, spínače a invertory. Jsou základem všech dnešních integrovaných obvodů, jako např. procesorů, pamětí atd.

TIP31 tranzistory jsou označeny jako TIP31A, TIP31B, TIP31C, TIP31.

Pro svoji práci jsem použil tranzistory TIP31C.

Parametry:

Obr. 5 Tranzistor TIP32

* Napětí: 115 V
* Proud: 3 A
* Výkon: 40 W

## Software

ArduinoIDE (IDE- Integrated Development Environment) obsahuje textový editor pro psaní kódu, textovou konzoli, nástrojovou lištu s tlačítky pro společných funkcí a řadou nabídek. To se připojí k hardwaru, nahraje programy a komunikuje s nimi. Programy jsou zapsány v textovém editoru a jsou uloženy s příponou souboru .ino. Poskytuje zpětnou vazbu při ukládání a exportu a také zobrazuje chyby.



Obr. 6 Logo arduino IDE

## Fritzing

##### Fritzing je open-source program pro tvorbu schémat zapojení. Obsahuje velké množství elektronických komponentů, se kterými můžeme pracovat. Pokud v programu nějaká součástka není, je možné ji do knihovny programu přidat. Součástí je také navrhování fyzického rozmístění komponentů na plošném spoji. Výsledné schéma můžeme uložit jako obrázek nebo jako šablonu pro vyleptání plošného spoje.



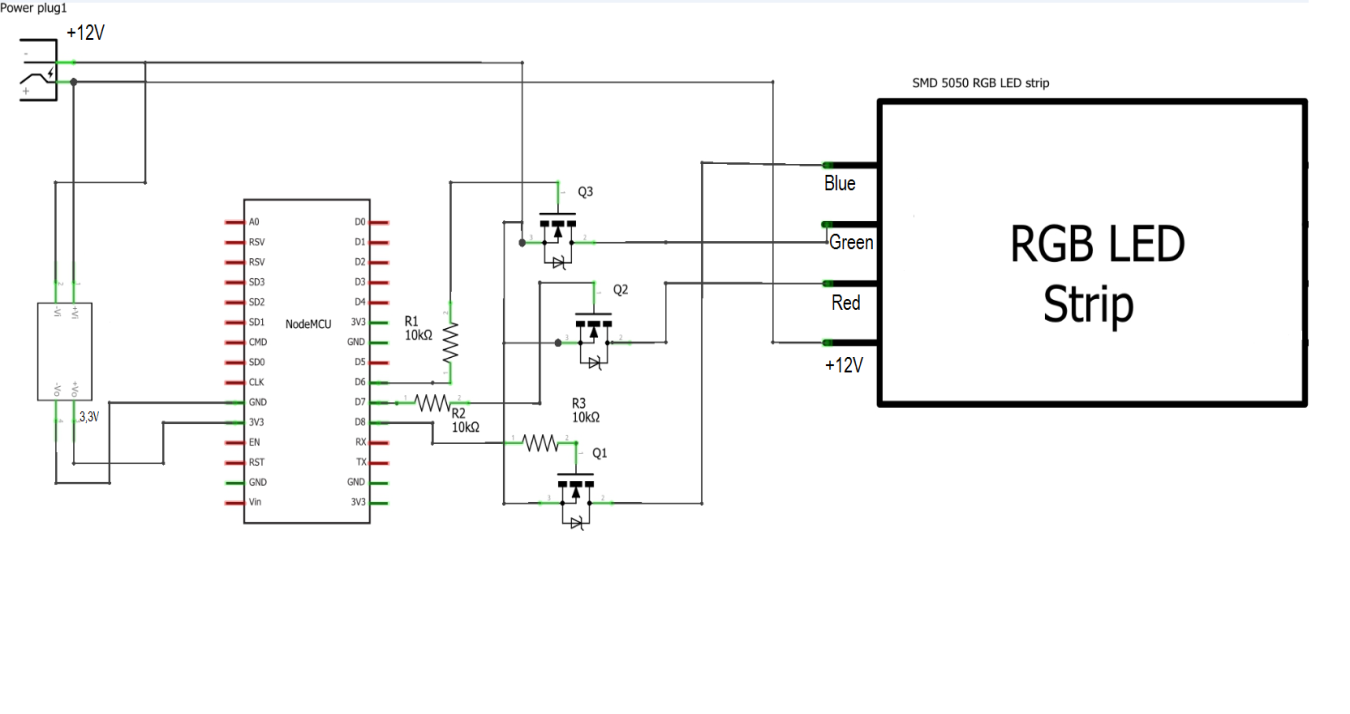
Obr. 7 Fritzing

# ZPŮSOBY ŘEŠENÍ A POUŽITÉ POSTUPY

## SEZNAM NUTNÝCH SOUČÁSTEK

* ESP8266 nodeMCUV3
* LED pásek SMD 5050
* Konvertor 12 V na 3,3 V
* Adaptér na 12 V
* 3x Tranzistor TIP31C
* 3x rezistor 10 KΩ
* Nepájivé kontaktní pole (později pájivé pole)

## SCHÉMA ZAPOJENÍ



Obr. 8 Schéma zapojení

## Ukázka kódu

V této části kódu se nastaví sériová linka, která poté vypíše odpočet a nastartuje se WiFi Access point, ke kterému se poté můžeme připojit telefonem nebo notebookem. Sériová linka poté vypíše IP adresu, na kterou se připojíme přes prohlížeč a můžeme ovládat LED pásky.

void setup() {

USE\_SERIAL.begin(115200);

//Cyklus pro odpočet

for (uint8\_t t = 4; t > 0; t--) {

USE\_SERIAL.printf("[SETUP] BOOT WAIT %d...\n", t);

USE\_SERIAL.flush();

delay(1000);

}

pinMode(BUTTON, INPUT\_PULLUP);

//přerušení, definování pinu přerušení

attachInterrupt(digitalPinToInterrupt(BUTTON), turnoff, HIGH);

pinMode(LED\_RED, OUTPUT);

pinMode(LED\_GREEN, OUTPUT);

pinMode(LED\_BLUE, OUTPUT);

digitalWrite(LED\_RED, 1);

digitalWrite(LED\_GREEN, 1);

digitalWrite(LED\_BLUE, 1);

WiFi.softAP("manin-led", "12345678"); //Nastavení jména a hesla WiFi

IPAddress myIP = WiFi.softAPIP();

USE\_SERIAL.print("AP IP address: ");

USE\_SERIAL.println(myIP);

## Webový server a webový klient

Webová stránka je jednou z částí mého projektu. Použil jsem jednoduchý webserver, kdy je uložena celá webová stránka jako řetězec a ta je poté serverem posílána klientovi, který ji pomocí webového prohlížeče zobrazí. Během spouštění aplikace musí dojít ke spuštění webserveru. Poté už pouze dochází k obsluze klienta.

## Webový server

V této části kódu pošleme na server HTML stránku (index.html), ve kterém si pak libovolně můžeme napsat webovou aplikaci, ve které můžeme měnit CSS styly.

server.on("/", []() {

// send index.html

// replace for better looking

server.send(200, "text/html",

" <html><head><script>"

....

"</body></html>");

});

## Ukázka HTML

Vytvoření webové stránky. Na stránce se nachází tlačítka pro změnu barvy a také posuvníky, pomocí kterých můžeme měnit jednotlivé složky barev: červená, zelená a modrá. Nastavují se hodnoty RGB od 0 do 255. Posuvníky můžou nastavit na LED páskách jakoukoliv barvu mimo černé (tzn. vypnutí LED pásků).

<h1 style=\"text-align: center;\">Ovládání RGB LED pásků</h1><br/>

<button onclick="zmena()" class="myButton" style="background-color:#f9f9f9;">náhodná barva</button>

<button onclick="on()" class="myButton" style="background-color:#f9f9f9;">bílá</button>

<button onclick="off()" class="myButton" style="background-color:#bfbfbf; color:#000000;">černá</button>

<button onclick="cer()" class="myButton" style="background-color:#ff8080; color:#800000">červená</button>

<button onclick="zel()" class="myButton" style="background-color:#99ff99; color:#1a6600">zelená</button>

<button onclick="mod()" class="myButton" style="background-color:#80bfff; color:#0000cc">modrá</button>

<button onclick="tyr()" class="myButton" style="background-color:#99ffff; color:#008080">tyrkysová</button>

<button onclick="ruz()" class="myButton" style="background-color:#ffb3ec; color:#800040">růžová</button>

<button onclick="zlu()" class="myButton" style="background-color:#ffffb3; color:#4d4d00">žlutá</button>

<input id=\"r\" type="range" min="0" max="255" step="1" onchange="sendRGB();" /><br/>

<input id=\"g\" type="range" min="0" max="255" step="1" onchange="sendRGB();" /><br/>

<input id=\"b\" type="range" min="0" max="255" step="1" onchange="sendRGB();" /><br/>

## Napájení

Jelikož jsem nechtěl, aby zařízení mělo 2 napájení, tak jsem musel vyřešit, aby přes jeden adaptér, který je na 12 V, šlo taky převést napětí na 3,3 V. K tomu jsem použil modul, kterému transformuje vstupní napětí 12 V na výstupní napětí 3,3 V. Poté vodiči přivedu správné napětí na správné piny

# **Závěr**

Cílem tohoto projektu bylo vytvoření funkčního kontroléru. Určené cíle, jako sestavení plošného spoje, vytvoření programu pro ovládání pásku a webové aplikace, byly splněny.

Do budoucna by bylo možné vytvořit webovou aplikaci, kde by byla paleta barev a bylo by možné si vybrat jakoukoliv barvu bez posuvníků.

Tento kontrolér jsem navrhl pro vlastní potřebu. Podobný kontrolér lze koupit za1000 Kč a já jsem si ho navrhl a sestrojil za čtvrtinu ceny.

Seznam použitýCH INFORMAČNÍCH ZDROJů

[1] Aritro Mukherjee. Getting Started With ESP-12E NodeMcu V3 . *C Sharp Corner.*[online]. 11.3.2016 [cit. 2017-01-10]. Dostupné z: [*http://www.c-sharpcorner.com/article/blinking-led-by-esp-12e-nodemcu-v3-module-using-arduinoide/*](http://www.c-sharpcorner.com/article/blinking-led-by-esp-12e-nodemcu-v3-module-using-arduinoide/)

[2] Jake Bergamin. Make A WiFi Controllable LED Strip. *Jake Bergamin.*[online]. 13.2.2016 [cit. 2017-01-10]. Dostupné z: *http://jakebergamin.com/2016/02/08/wifi-led-strip/*

[3] NodeMCU – Wikipedia. *Wikipedia.*[online]. 3.1.2017 [cit. 2017-01-10]. Dostupné z: [*https://en.wikipedia.org/wiki/NodeMCU*](https://en.wikipedia.org/wiki/NodeMCU)

[4]Krzysztof. Arduino/soft-access-point-examples.md at master esp8266/Arduino GitHub. *GitHub.*[online]. 25.8.2016 [cit. 2017-01-10]. Dostupné z: *https://github.com/esp8266/Arduino/blob/master/doc/esp8266wifi/soft-access-point-examples.md*

Seznam příloh

č. 1 Aplikace

č. 2 Schéma

č. 3 Fotodokumentace