|  |  |  |
| --- | --- | --- |
| Logo__SSPU_2016_Barva | | |
| **Závěrečná studijní práce**  **dokumentace** | | |
| **Laserová závora** | | |
| Jakub Klíštinec | | |
| https://i.ytimg.com/vi/xS3ipW3-cMU/hqdefault.jpg | | |
|  | |  |
| **Obor:** | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| **Třída:**  **Školní rok:** | IT4  2018/2019 | |

##### **Poděkování**

*Chtěl bych poděkovat panu Ing. Petru Grussmannovi za pomoc při řešení určitých problémů a panu Ing. Jiřímu Miekischovi za konzultace. Rovněž musím poděkovat svému strýci za pomoc se sháněním správných součástek.*

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

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

V Opavě 31. 12. 2018

*podpis autora práce*

**ANOTACE**

Laser svítící na odrazku a z odrazky následně na fototranzistor slouží jako bezpečné chráněné území, jejímž výsledkem je zmenšenina jedné místnosti. Laser je napájen na 6V z externího zdroje a je ovládán Wi-Fi mikročipem D1 Mini Pro ESP8266-12, který vysílá přerušovaný signál tak, aby zamezil jakémukoliv zneužití a možnému vniknutí. Vyhodnocování tohoto signálu nemá na starosti pouze již zmiňovaný mikročip, ale také komparátor. V případě jakéhokoliv narušení bezpečnosti je okamžitě informován vlastník skrze odesílání dat na internet za pomoci MQTT serveru. Program pro bezchybnou funkci tohoto projektu byl napsán v programu zvaném Visual Studio Code s prostředím PlatformIO IDE v programovacím jazyce C++.

**Klíčová slova**

Laser, Fototranzistor, ESP 8266, přerušovaný signál, MQTT server, PlatformIO IDE.

OBSAH

[Úvod 6](#_Toc534715603)

[Využité součástky a software 7](#_Toc534715604)

[1.1 Klíčová komponenta 7](#_Toc534715605)

[1.1.1 ESP D1 Mini Pro 7](#_Toc534715606)

[1.2 Vysílač a přijímač 7](#_Toc534715607)

[1.3 VS Code a PlatformIO 7](#_Toc534715608)

[1.4 Ostatní součástky 8](#_Toc534715609)

[1.4.1 Rezistory 8](#_Toc534715610)

[1.4.2 Piezoelektrický bzučák 8](#_Toc534715611)

[1.4.3 Komparátor 9](#_Toc534715612)

[1.4.4 Trimr 9](#_Toc534715613)

[1.4.5 Tlačítko 10](#_Toc534715614)

[2 Postupy řešení 11](#_Toc534715615)

[2.1 Zapojení 11](#_Toc534715616)

[2.1.1 Schéma 11](#_Toc534715617)

[2.2 Ukázky kódu 12](#_Toc534715618)

[3 ŘEŠENÍ PROBLÉMŮ 13](#_Toc534715619)

[3.1 Linux 13](#_Toc534715620)

[3.1.1 VirtualBox 13](#_Toc534715621)

[3.2 Nekvalitní čip 13](#_Toc534715622)

[3.2.1 LoLin NodeMCU 13](#_Toc534715623)

[3.3 Nezvládnuté postupy 14](#_Toc534715624)

[3.3.1 Přerušovaný signál 14](#_Toc534715625)

[3.3.2 Odesílání dat 14](#_Toc534715626)

[4 Konečná úprava 15](#_Toc534715627)

[4.1 Vzhled 15](#_Toc534715628)

[Závěr 16](#_Toc534715629)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 17](#_Toc534715630)

[Seznam příloh 19](#_Toc534715631)

Úvod

Cílem tohoto projektu bylo vytvořit chráněný prostor za pomocí jednoduchých a levných součástek, které jsou běžně k sehnání. Například laser byl vyndán z malé propisovací tužky a následně lehce upraven. Ostatní součástky jsou rovněž velice levné. Jediný problém je se sehnáním kompárátoru, který se shání trochu hůře, bylo by teda lepší jej dopředu objednat přes internet.

Ze začátku bylo mou hlavní prioritou udělat jednoduchou závoru, která spustí alarm v případě narušení. Později pak přicházela postupná vylepšení, jako například možnost informovat vlastníka v případě narušení bezpečnosti či přerušovaný signál, který způsobuje, že jen a pouze jedna frekvence z vysílače je přijímána. Rovněž jsem byl nucen změnit mikročip za jiný, abych s pomocí konzole zjistil, jak fototranzistor běží a jak snímá světlo.

Ve své dokumentaci se hned ze začátku budu věnovat podrobněji součástkám a také softwaru. Následně detailně přiblížím své zapojení a některé zajímavé části mého kódu. Ve třetí části se budu věnovat řešeným problémům, následně vzhledu a v samotném závěru budu věnovat pozornost výsledkům mého snažení.

# Využité součástky a software

## Klíčová komponenta

### ESP D1 Mini Pro



Obrázek 1: ESP D1 Mini Pro

Tento typ ESP umí na své rozměry spoustu věcí a funguje bezchybně a rychle. Pro spojení se zařízením stačí použít normální USB kabel. Má zde zabudovanou anténu a rovněž umí odeslat z jednoho pinu až 5V. Jinak však přijímá 3,3V jako drtivá většina všech ostatním mikročipů z řad ESP 8266.

## Vysílač a přijímač

Jako vysílač jsem použil laser vytažený z propisovací tužky, který stál asi 10Kč. Je napájený přímo z ESP na 3,3V. Přijímačem je fototranzistor typu LTR-3208, který bez problémů stačí s maximem až do 30V na využití u tohoto projektu.

## VS Code a PlatformIO

Visual Studio Code je volně dostupný neplacený editor, v kterém může uživatel programovat. Zajímavější je však vývojové prostředí, kterému říkáme PlatformIO. Je opět volně stažitelné na oficiálních stránkách PlatformIO a je podporované většinou operačních systémů. Je velmi výhodné, že je zde možnost kompilovat pro více druhů desek. Můžete také programovat v C++ a celé prostředí se chová velmi chytře, protože napovídá uživateli, co zřejmě chce zadat.

## Ostatní součástky

### Rezistory



Obrázek 2: Rezistor

Rezistory jsou zapotřebí snad úplně všude. Nejčastěji jsem zde používal 300Ω anebo 1 a 10 kΩ.

### Piezoelektrický bzučák



Obrázek 3: Piezoelektrický bzučák

Tuto součástku jsem použil jako výstražné zařízení, které upozorní na přerušení paprsku. V testovacích verzích je dobré si tuto součástku nahradit například LED diodou, protože tyto zvuky z bzučáku nejsou příliš příjemné.

### Komparátor



Obrázek 4: Komparátor

Toto zařízení porovnává dvě hodnoty a vyhodnocuje přerušovaný signál a podle toho pozná, jestli na přijímač svítí správné světlo, či ne. Tato součástka, která nese označení typu LM393N, má dohromady 8 pinů a zdá se být složitá. Je tomu tak, protože v této součástce je možné komparátory vytvořit dva. V mém případě stačí pouze jeden. Využijete tedy pouze pět pinů a to: zemi, kladné napětí (obě jsou pro oba komparátory stejné) a následně dvě hodnoty, které chcete porovnávat a poslední vrací hodnoty do mikročipu.

### Trimr



Obrázek 5: Trimr

Trimr je rezistor s proměnným odporem, jednoduše nastavitelný rezistor. V mém případě jsem trimr 500Ω použil jako dělič napětí. Funguje tak, že šroubovákem jednoduše potočíte na vršku součástky.

### Tlačítko



Obrázek 6: Tlačítko

Nemohu zapomenout také na tlačítko, které se hodí v případě nějakého restartu či vypnutí alarmu.

Rovněž jsou zapotřebí propojovací vodiče, když pracujete na testech na nepájivém kontaktním poli.

# Postupy řešení

## Zapojení

### Schéma

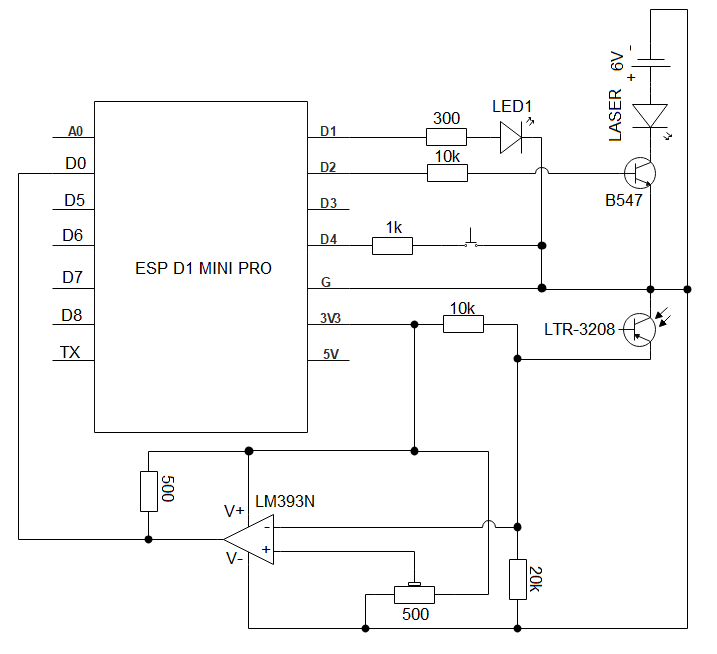


Schéma bylo nakresleno v programu EDraw Max.

LTR-3208 … Fototranzistor

LM393N … Komparátor

B547 … Tranzistor typu NPN

Pomocí tranzistoru je spínáno za pomoci externího zdroje napětí šesti voltů, kvůli dostatečnému napájení laseru. V konečném výsledku je LED1 nahrazena piezoelektrickým bzučákem.

## Ukázky kódu

Tato část kódu je zaměřena na první pokusy s fototranzistorem, kdy ještě data nevyhodnocoval komparátor. Fototranzistor v mém případě funguje tak, že snímá světlo od hodnot 0 – 1024 (čím více světla, tím menší číslo). Na testech jsem zjistil, že v normálně osvětlené místnosti udává fototranzistor hodnoty okolo 850 – 950. (to znamená, že na přijímač skoro žádné světlo přímo nesvítí). Stanovil jsem teda jednoduchou podmínku, že když bude hodnota pod 600 (tj. laser výrazněji svítí na fototranzistor, vypíše se mi to do konzole a LED dioda zůstane vypnutá, neboť prostor není narušen. Když laser svítí přímo na fototranzistor, udává obvykle hodnoty okolo 50. Jakmile nesvítí nic, spustí se druhý blok příkazů, který se nachází pod příkazem „else“. Tam nechybí opět výpis do konzole, rozsvícení ledky a nastavení proměnné „trigger“ typu boolean na hodnotu false. Ta je zde kvůli druhé části kódu a tím je resetovací tlačítko. To zde funguje tak, že když laser na přijímač nesvítí, zapne se LED dioda. Poté odejmu překážku a laser opět svítí na fototranzistor, avšak LED dioda zůstane vypnutá. Proto je nutné stisknout tlačítko v moment, kdy už mezi vysílačem a přijímačem není žádná překážka a LED dioda se opět vypne.

void loop()

{

ldr\_value = analogRead(ldr); //čte hodnoty LDR

Serial.println(ldr\_value); //zobrazí hodnoty LDR na sériové lince

if (trigger){

if (ldr\_value < 600) {

Serial.println("mensi nez 600");

//digitalWrite(LED, LOW);

}

else {

Serial.println("vetsi nez 600");

trigger = false;

digitalWrite(LED, HIGH);

}

}

else{

if(digitalRead(BUTTON) == LOW){

trigger = true; //tlačítko je sepnuté

digitalWrite(LED, LOW);

Serial.println("zmackl jsi tlacitko");

}

else{

Serial.println("Nezmackl jsi tlacitko");

}

}

delay(500);

}

# ŘEŠENÍ PROBLÉMŮ

## Linux

### VirtualBox

Ne každý má po ruce notebook s možností nahrát si na něj Linux a začít pracovat, tak jsem se rozhodl, že to zkusím nejprve ve VirtualBoxu. Zprvu to šlo hladce. Nainstaloval jsem editor a vývojové prostředí. Následně napsal jednoduchý kód, ale hlásil mi spoustu chyb. Některé z nich jsem vyřešil, jiné však nikoli. Po asi 14 dnech jsem své snažení vzdal a požádal o pomoc pana učitele. Ani po konzultacích jsem však nebyl úspěšný a od tohoto postupu jsem musel upustit.

## Nekvalitní čip

### LoLin NodeMCU



Obrázek 7: ESP LoLin NodeMCU

Původně jsem začínal s právě tímto typem ESP. Fungovalo správně a spoustu věcí jsem si na něm odzkoušel. Avšak po měsíci, kdy už jsem se dostal k fototranzistoru a měření hodnot, potřeboval jsem vidět výpis, co mi to hlásí. Stejný problém bych měl později, když bych měl složitější kód, tak je dobré si do každých částí psát nějaké hlášky, abyste přesně věděli, kde program ztroskotal. A právě tento mikročip to neuměl nebo jsme na to s panem učitelem nepřišli. Po několika týdnech jsem byl nucen začít pracovat s jiným mikročipem a pan učitel byl ochoten mi poskytnout jiný, právě již výše zmíněný D1 Mini Pro.

## Nezvládnuté postupy

### Přerušovaný signál

V současné době mám kód napojený na komparátor, který porovná dvě hodnoty a podle toho rozhodne, co dál, což je správně. Nyní jsem chtěl z ESP vysílat přerušovaný signál, avšak nemohu svůj kód zkompilovat z neznámých důvodů. Podle chyb, které mi byly hlášeny jsem se dočetl pouze to, že by to mohlo mít souvislost s hard diskem mého starého notebooku. Podrobné návody na odstranění problému se mi zatím nepodařily vyřešit.

### Odesílání dat

Jako úkol navíc jsem dostal Homeassistant a odesílání dat na internet či na email v případě, že je prostor narušen. Z časových důvodů a neustálých trablí s PlatformIO a Linuxem jsem se k tomuto kroku zatím vůbec nedostal.

# Konečná úprava

## Vzhled

Obrázek

Text

# **Závěr**

Cílem projektu bylo vytvořit zabezpečovací systém pro místnost za levné peníze, který by v případě narušení bezpečnosti zapnul alarm. V současné době se ladí poslední detaily ohledně přerušovaného signálu. Bohužel s největší pravděpodobností nebude k dispozici odesílání dat na internet.

Do budoucna mám rozhodně plány tento projekt zcela dokončit. Začal bych rozhodně již zmíněným odesíláním dat na internet. Jednoduché základy už v tom mám, protože jsem to ve škole několikrát zkoušel. Chtěl bych, aby mi data přicházela na email. Následně bych vše zapájel na dvě desky, ať to taky nějak vypadá z blízky a ať se s celým zařízením lépe manipuluje. Poté bych se vrhl do opravdu reálného využití a to tak, že bych koupil silnější laser, upravil napájení a alarm bych mohl umístit například do garáže. V současné době se domnívám, že by to tento laser a ještě k tomu s daným slabším napájením nezvládl.

Seznam použitýCH INFORMAČNÍCH ZDROJů

[1] Arduino s fotorezistorem (světelný sensor). [online] [cit. 2019-01-06].

<http://arduinonavody.cz/arduino-s-fotorezistorem-svetelny-senzor/>

[2] Why is apt-get always using proxy [online] [cit. 2019-01-06].

<<https://askubuntu.com/questions/344802/why-is-apt-get-always-using-proxy-although-no-proxy-is-configured>>

[3] Homie dual relay [online] [cit. 2019-01-06].

<https://github.com/petrgru/homie-dual-relay>

[4] Laser data transfer [online] [cit. 2019-01-06].

<https://www.instructables.com/id/Laser-data-transfer-part2-Binary-transmitting//>

[5] Laserová závora – nová verze [online] [cit. 2019-01-06].

<https://www.gme.cz/data/attachments/orn.760-492.1.pdf>

[6] Úvodní obrázek - laser [online] [cit. 2019-01-06].

<https://i.ytimg.com/vi/xS3ipW3-cMU/hqdefault.jpg>

[7] Obrázek 1 [online] [cit. 2019-01-06].

< http://www.neven.cz/user/7262/upload/stuff/17892536.jpg>

[8] Obrázek 2 [online] [cit. 2019-01-06].

< https://www.gme.cz/data/product/1024\_1024/pctdetail.110-097.1.jpg

[9] Obrázek 3 - laser [online] [cit. 2019-01-06].

< <https://www.vpcentrum.eu/image/cache/data/prods/prod7/290677-piezobzucak-13-8mm-fy-14-napajeni-3-18v-7ma-1-500x500.jpg>>

[10] Obrázek 4 – LM393N [online] [cit. 2019-01-06].

< https://www.ges.cz/cz/lm393n-GES05001247.html>

[11] Obrázek 5 – Uhlíkový trimr PT10MVE500 [online] [cit. 2019-01-06].

< https://www.gme.cz/pt10mve500>

[12] Obrázek 6 – Mikrospínač 6x6x6 mm tlačítko [online] [cit. 2019-01-06].

< https://arduino-shop.cz/arduino/1253-mikrospinac-6x6x6-mm-tlacitko-1459621295.html?gclid=EAIaIQobChMI6ev7x9zZ3wIVAud3Ch0\_twKPEAQYCSABEgI-MPD\_BwE>

[13] Obrázek 7 [online] [cit. 2019-01-06].

< https://www.aerial.net/shop/imageslarge/IOT-Lolin\_main.jpg>

Seznam příloh

č. 1 Titulní list

č. 2 Čestné prohlášení

č. 3 Poděkování

Nepovinná část – pokud nemáte žádné přílohy ke své práci, tuto část odstraňte!

* Přílohy se zařazují na konec práce.
* Jsou to texty, obrázky, grafy, tabulky, které by přímo v textu byly zbytečně detailní, ale mají být po ruce k dokreslení východisek i výsledku řešení.
* Jsou číslovány a v textu se na ně může odkazovat.
* Před první přílohu se umisťuje seznam příloh.
* Každá příloha je označena číslem - např. Tabulka č.. 1, Schéma č. 2, Obrázek č. 3.
* Každá tabulka by měla mít i vlastní název, který stručně vystihuje její obsah.
* (Tabulka č. 1 Zakázky stavebních prací v roce 2009-2010).
* Pokud je z tabulky vytvořen graf, umístíme jej na stejné stránce jako tabulku.

**Příloha č. 1: Titulní list**