



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ

FACULTY OF MECHANICAL ENGINEERING

ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY

INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

DÁLKOVÝ SYSTÉM PRO ZÁŽEH ZÁBAVNÍ PYROTECHNIKY

REMOTE SYSTEM FOR IGNITION OF FIREWORKS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Dominik Štrúbel

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Formánek

BRNO 2023

Zadání bakalářské práce

Ústav:	Ústav mechaniky těles, mechatroniky a biomechaniky
Student:	Dominik Štrúbel
Studijní program:	Aplikované vědy v inženýrství
Studijní obor:	Mechatronika
Vedoucí práce:	Ing. Martin Formánek
Akademický rok:	2022/23

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

Dálkový systém pro zážeh zábavní pyrotechniky

Stručná charakteristika problematiky úkolu:

Práce se bude zabývat návrhem a sestavením systému pro bezpečný dálkový zážeh zábavní pyrotechniky. Systém bude obsahovat vysílač a více přijímačů, každý z přijímačů bude mít více kanálů (jeden přijímač bude schopen zážehu více zápalnic). Pomocí vysílače bude možný zážeh po jednotlivých kanálech nebo nastavení časových sekvencí. Typ odpalu bude možno zvolit v uživatelském programu. Zážeh bude dostatečně zřetelně signalizován zvukovou a/nebo světelnou signalizací. Dbáno bude hlavně na bezpečnost a spolehlivost systému.

Cíle bakalářské práce:

1. Provedte rešerši v oblasti dostupných řešení, průzkum trhu a výběr vhodných komponent automatizovaného zážehu zábavní pyrotechniky.
2. Navrhněte a realizujte vlastní řešení .
3. Vytvořte program pro zadávání sekvencí zážehu.
4. Ověřte funkčnost navrženého řešení a posuďte jej z hlediska bezpečnosti.

Seznam doporučené literatury:

[1] Sen Gupta, Gourab & Mukhopadhyay, S.C.. (2010). Embedded Microcontroller Based Fireworks Detonation System. 10.1007/978-3-642-13636-8_8.

[2] Santos, R.A. & Block, A.E.. (2012). Embedded systems and wireless technology.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2022/23

V Brně, dne

L. S.

prof. Ing. Jindřich Petruška, CSc.
ředitel ústavu

doc. Ing. Jiří Hlinka, Ph.D.
děkan fakulty

Abstrakt

Bakalářská práce se zabývá návrhem a realizací vlastního programovatelného bezdrátového systému pro zážeh zábavní pyrotechniky. Rešeršní část se zabývá průzkumem dostupných řešení a technologiemi použitých k realizaci vlastního řešení, včetně použité technologie LoRa. Praktická část práce se zabývá návrhem DPS a konstrukce. Následným sestavením systému, naprogramování jeho funkcí a vytvoření počítačové aplikace pro nastavení sekvencí zážehů.

Summary

The bachelor thesis focuses on design and implementation of own programmable wireless system for fireworks ignition. The research part focuses on the research of available solutions and technologies used to implement the own solution, including the used technology LoRa. The practical part of the thesis focuses on the design of the PCB and the construction. Then, building the system, programming its functions and creating a computer application for setting the ignition sequences.

Klíčová slova

zábavní pyrotechnika, dálkový zážeh, bezdrátový přenos, LoRa, Arduino, Python 3

Keywords

fireworks, remote ignition, wireless transmission, LoRa, Arduino, Python 3

Bibliografická Citace

ŠTRÚBEL, D. *Dálkový systém pro zážeh zábavní pyrotechniky*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2023. 46 s., Vedoucí diplomové práce: Ing. Martin Formánek.

Čestné prohlášení

Prohlašuji, že tato práce je mým původním dílem, zpracoval jsem ji samostatně pod vedením Ing. Martina Formánka a s použitím literatury uvedené v seznamu.

Dominik Štrúbel

Brno

.

Poděkování

Touto cestou bych chtěl poděkovat hlavně vedoucímu mojí bakalářské práce Ing. Martinu Formánkovi za jeho přístup a věcné připomínky při vedení této práce. V neposlední řadě bych rád poděkoval rodině a přítelkyni za podporu během studia.

Dominik Štrúbel

Obsah

1	Úvod	8
2	Rešerše	9
2.1	Dostupná řešení na trhu	9
2.2	Možnosti pro návrh vlastního řešení	12
2.2.1	Bezdrátová komunikace	12
2.2.2	Mikrokontrolér	14
2.2.3	Napájení a nabíjení	14
2.3	Python	16
2.3.1	pySerial	16
2.3.2	Tkinter	16
3	Postup a výsledky řešení	18
3.1	Popis funkce systému	18
3.1.1	Požadavky na systém	20
3.2	Sestavení prototypu	21
3.2.1	Návrh DPS	21
3.2.2	Výroba krabiček	24
3.2.3	Kompletace řešení	24
3.3	Programování	28
3.3.1	Firmware pro vysílač a přijímač	28
3.3.2	Počítačový program pro zadávání sekvencí	31
3.4	Ověření funkčnosti návrhu	34
3.4.1	Ověření funkčnosti PC programu	34
3.4.2	Spolehlivost bezdrátové komunikace	34
3.4.3	Posouzení z hlediska bezpečnosti	34
3.4.4	Zkouška výdrže baterie	34
4	Závěr	36
	Literatura	37
	Seznam zkratk	41
	Seznam příloh	43

1 Úvod

Zábavní pyrotechnika patří mezi velmi oblíbenou zábavu při slavnostech a oslavách po celém světě. Bohužel s touto zábavou přichází také vyšší riziko nehod pro lidi, kteří se pohybují v blízkosti pyrotechniky, především pokud nejsou dodržena všechna důležitá bezpečnostní opatření. Ke spoustě nehod, spojených s používáním zábavní pyrotechniky, dochází při silvestrovských oslavách, kdy jednotlivci často nezacházejí s pyrotechnikou zodpovědně a bezpečně. Proto jsem se rozhodl zabývat vytvořením bezpečného a spolehlivého dálkového systému pro zážeh zábavní pyrotechniky.

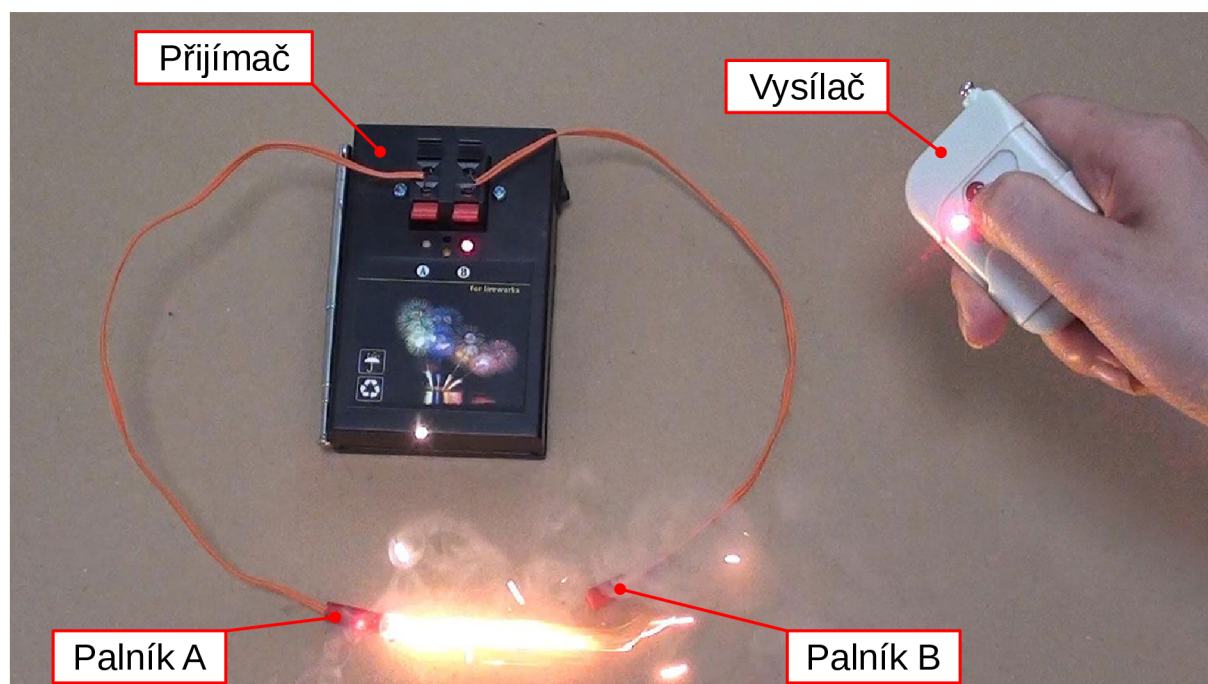
V rešerši této práce jsem se zaměřil na průzkum v oblasti dostupných řešení systémů pro dálkový zážeh zábavní pyrotechniky a popis jejich funkce. Taktéž jsem se zaměřil na průzkum technologií a jejich využití při tvorbě takového systému.

Cílem práce je navrhnout, sestavit a otestovat bezpečný dálkový systém pro zážeh zábavní pyrotechniky, který by mohl být využíván v průběhu různých oslav. Tímto systémem bych chtěl minimalizovat rizika spojená s používáním zábavní pyrotechniky a umožnit lidem prožít oslavy ve zdraví bez obav o svou bezpečnost, se zážitkem hezkého ohňostroje.

2 Rešerše

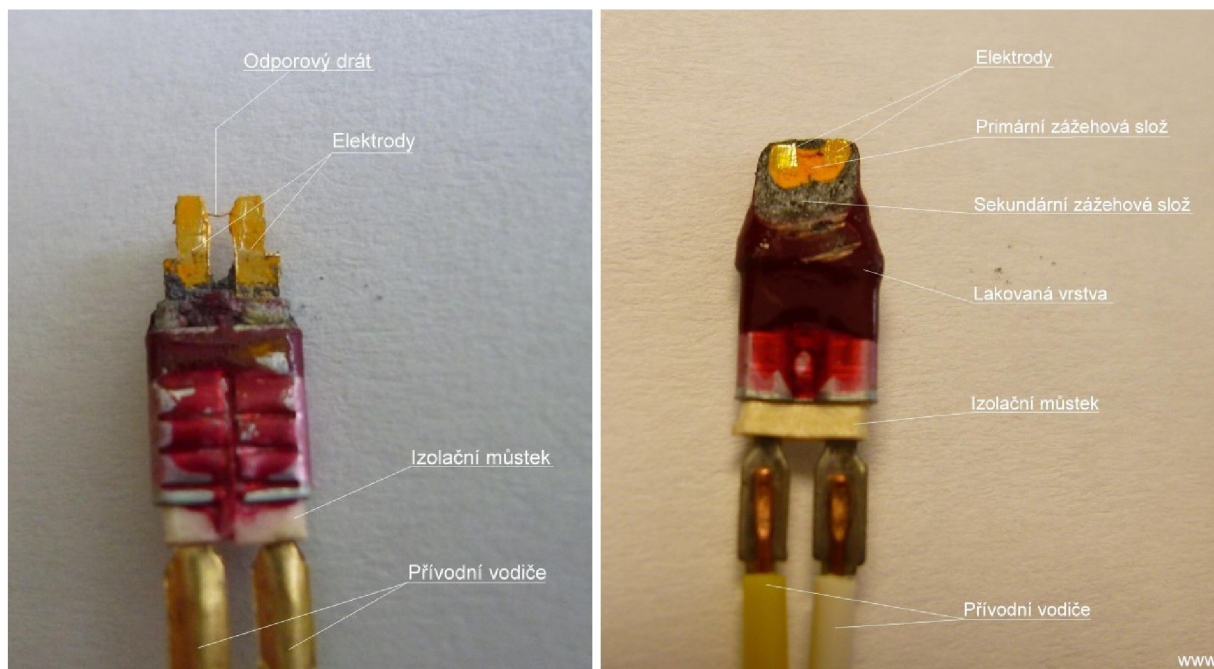
2.1 Dostupná řešení na trhu

Většina odpalovacích zařízení, běžně dostupných na českém trhu, má dle mého průzkumu velmi podobná provedení a pro zážeh využívá tzv. elektrický palník. Jak elektrický palník funguje a z jakých částí se skládá zde popíšu později. Při odpalu zábavní pyrotechniky připevníme tento palník k zápalnici pyrotechniky a připojíme ke konektoru odpalovacího zařízení, zařízení zapneme a vzdálíme se do bezpečné vzdálenosti, kde můžeme zábavní pyrotechniku pomocí vysílače odpálit. Na obrázku 2.1 je vyobrazeno odpalovací zařízení s připojenými palníky v okamžiku kdy došlo k zážehu palníku připojeného na výstupu „A“, po stisknutí tlačítka „A“ na vysílači [1, 2].



Obrázek 2.1: Zážeh palníku pomocí dálkově ovládaného odpalovacího zařízení [1]

Elektrický palník používaný k zážehu zábavní pyrotechniky je v podstatě odporový drát, umístěný na koncích vodičů, obalený citlivou pyrotechnickou složí. Průchodem elektrického proudu se odporový drát zahřeje, zažehne pyrotechnickou slož a ta následně zapálí zápalnici zábavní pyrotechniky. Na obrázku 2.2 jsou podrobně popsány jednotlivé části elektrického palníku. Rozlišuje se mezi dvěma druhy palníků, středně odolné a nízko odolné. Odpor středně odolného palníku se pohybuje v rozmezí od 0,48 do 0,56 Ω a jeho bezpečný proud je maximálně 0,45 A po dobu pěti minut. Odpor nízko odolného palníku se pohybuje v rozmezí od 1,5 do 1,9 Ω a jeho bezpečný proud je maximálně 0,18 A po dobu pěti minut. K zážehu palníku je zapotřebí proudový impuls o délce 4 ms a velikosti 2,12 A



Obrázek 2.2: Popis konstrukce elektrického palníku [3]

pro středně odolný palník a o velikosti 1 A pro nízko odolný palník. V případě odpalu vícero palníků současně se nejběžněji využívá sériového zapojení. Při sériovém zapojení je sice zapotřebí vyššího napětí, ale zároveň zůstává zachován proud potřebný pro zážeh. Navíc lze takto snadněji odhalit případné chybné zapojení nebo vadné palníky [3, 4].

Sady prodávané na českém trhu většinou obsahují jednu nebo více přijímacích jednotek ve dvou provedeních. První provedení přijímacích jednotek, které je vidět na obrázku 2.3, má pouze jeden výstup pro připojení palníků. Druhé provedení, na obrázku 2.4, má již výstupy čtyři. Dále pak sady obsahují dálkový ovladač, který se mezi sadami liší počtem tlačítek. Na ovladači je ke každému tlačítku přiřazen jeden výstup z přijímacích jednotek [2, 5, 6, 7, 8, 9].



Obrázek 2.3: Sada odpalovacích zařízení prodejce Pyrostav.cz, obsahující vysílač a čtyři přijímače s jedním výstupem [6]



Obrázek 2.4: Sada odpalovacích zařízení prodejce CHILI ROSES.CZ, obsahující vysílač a dva přijímače se čtyřmi výstupy [2]

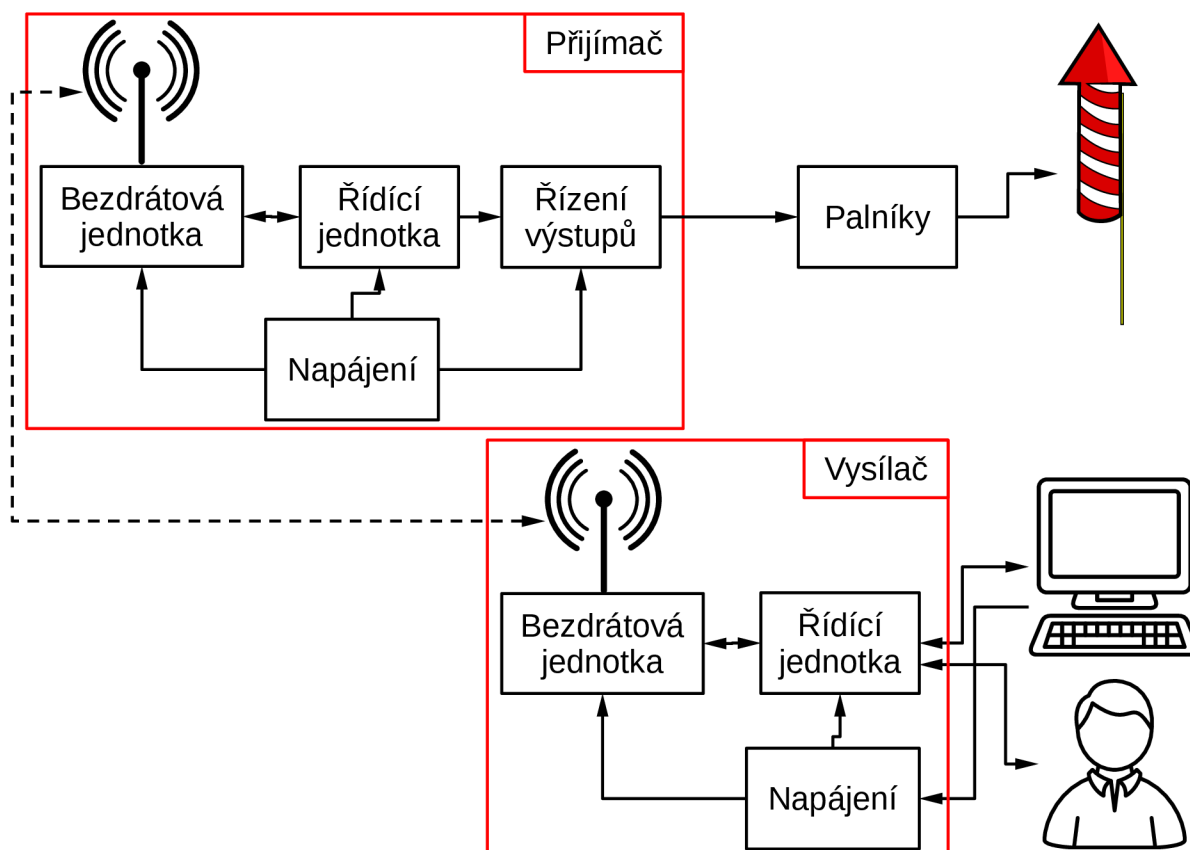
Všechny tyto sady, objevené v mém průzkumu trhu, jsou porovnány v tabulce 2.1. V této tabulce není uvedena frekvence používaná pro komunikaci mezi odpalovacími zařízeními a dálkovým ovladačem, jelikož všechna porovnávaná zařízení využívají ke komunikaci frekvenci 433 MHz. Některé sady, podle popisu prodejců, jsou kromě odpalu po zmáčknutí tlačítka také schopny odpalu všech výstupů najednou nebo v sekvencích, což se provádí současným stiskem dvou tlačítek na ovladači [2, 5, 6, 7, 8, 9].

Prodejce	Počet odpalů	Počet Přijímačů	Cena [Kč]	Dosah [m]	Napájení	
					Přijímač	Vysílač
CHILI ROSES.CZ	8	2	1 980	až 100	neuveďeno	neuveďeno
	24	6	4 200	150–300	neuveďeno	neuveďeno
ignisshop.com	12	3	3 305	až 400	4× AA	A23
Pyrostav.cz	1	1	920	200	4× AAA nebo 9V	A23
	4	4	2 490	200	4× AAA	A23
	12	12	7 150	200	4× AAA	A23

Tabulka 2.1: Porovnání dostupných řešení prodávaných na českém trhu [2, 5, 6, 7, 8, 9]

2.2 Možnosti pro návrh vlastního řešení

Při návrhu vlastního řešení jsem se snažil využít spolehlivých a snadno dostupných součástek. Jelikož volba součástek je důležitým krokem k vytvoření a oživení funkčního prototypu. Jak takový návrh prototypu může vypadat je vidět na obrázku 2.5.



Obrázek 2.5: Blokové schéma návrhu dálkového systému pro zážeh zábavní pyrotechniky

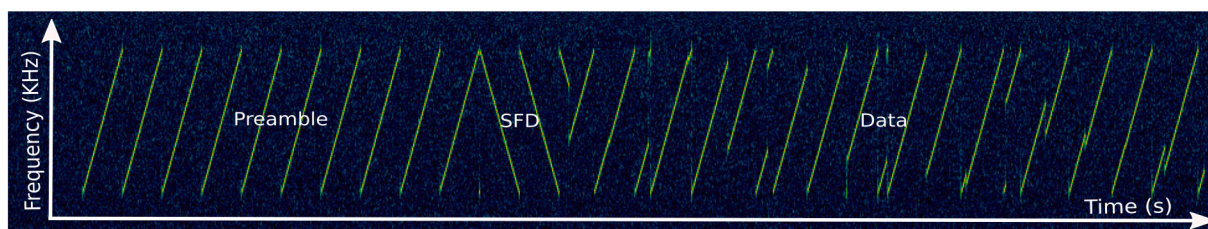
2.2.1 Bezdrátová komunikace

Pod pojmem bezdrátová komunikace si lze představit výměnu informací na dálku mezi dvěma nebo více uživateli bez použití spojení uživatelů metalickým, optickým nebo jiným kabelem [10]. Z historického hlediska nás bezdrátová komunikace doprovází již dlouhá léta od dob kdy se ke komunikaci na dálku využíval křik, bubny nebo kouřové signály. Nicméně bezdrátová komunikace tak jak ji dnes známe odstartovala na základě práce Maxwella a Herze, kteří položili základ pro pochopení přenosu elektromagnetických vln [11]. Přenos informací volným prostorem pomocí elektromagnetických vln s sebou přináší také rizika (např. rušení nebo odposlech). S rostoucí vzdáleností mezi přijímačem a vysílačem taktéž klesá intenzita přijímaného signálu. Velkou výhodou používání bezdrátové komunikace je však mobilita uživatele [10].

Existuje spousta technologií pro bezdrátovou komunikaci, ale ne všechny technologie je možné použít např. kvůli nedostatečnému dosahu. Proto jsem zvolil, v posledních letech rozšiřující se, technologii LoRa (Long Range) navrženou pro IoT (Internet of Things). Technologie LoRa dokáže komunikovat při nízkých výkonech na velké vzdálenosti i několik kilometrů a využívá nelicencovaná pásma ISM (Industrial, Scientific and Medical radio).

Vzdálenost, na kterou je LoRa schopna komunikovat, se liší podle nastavení, vysílacího výkonu a terénu prostředí v kterém jsou zařízení provozována [12].

LoRa využívá k modulaci signálů variantu modulace CSS (Chirp Spread Spectrum), která je dle společnosti Semtech odolná proti vysokému stupni rušení. Chirp v CSS označuje signál s konstantně se zvyšující nebo snižující frekvencí označovanou jako upchirp a downchirp, která se pohybuje v rozmezí šířky pásma. Na obrázku 2.6 jsou znázorněny různé typy chirpů. Začátek tvoří upchirpy u kterých se frekvence s časem zvyšuje od minimální po maximální frekvence. Následují downchirpy označené jako SFD (Start Frame Delimiter), které začínají na maximální frekvenci a s časem se snižují na minimální frekvenci. Poté již pokračují modulované chirpy, které obsahují datové bity [12].

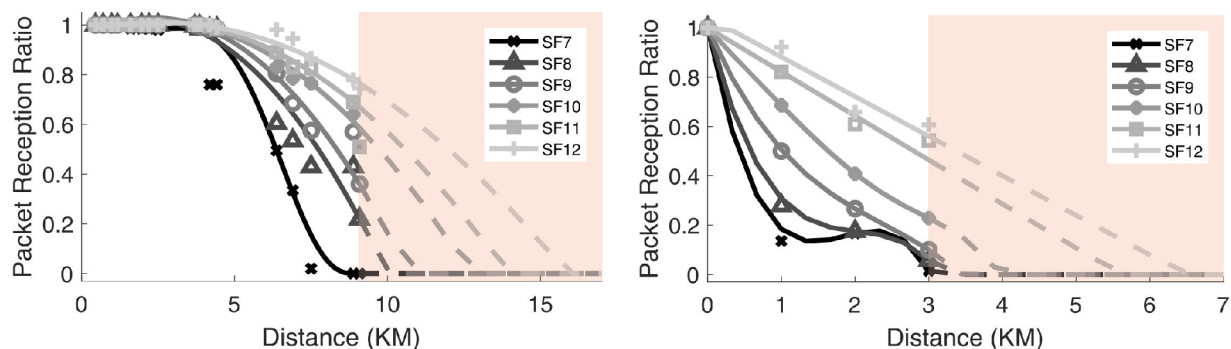


Obrázek 2.6: Snímek přenosu paketu LoRa [12]

Přenosovou vzdálenost ovlivňuje nastavení čtyř hlavních parametrů, které mají výrazný vliv na přenos signálu. Tyto parametry jsou:

- **Spreading Factor (SF)**
SF určuje kolik bitů je v jednom chirpu. Například při nastavení SF7 bude v jednom chirpu právě 7 bitů. S každým zvýšením SF o jeden stupeň se taktéž zdvojnásobuje čas potřebný k přenosu chirpu. Zdvojnásobení doby přenosu, má přijímač více možností ke vzorkování výkonu přijímaného signálu, což vede k vyššímu poměru SNR (Signal-to-Noise Ratio).
- **Bandwidth (BW)**
Změnou šířky pásma dochází ke změně doby trvání přenosu a tím i ke změně SNR.
- **Transmission Power (TX Pow)**
Vysílací výkon přímo ovlivňuje množství energie použité k přenosu a zvyšuje tak efektivní dosah vysílaného signálu.
- **Code Rate (CR)**
CR je přidávána do paketu před přenosem a slouží pro kontrolu správnosti přijatých dat a opravu chyb vzniklých při přenosu. [12]

Autoři textu z kterého jsem čerpal, uskutečnili experimentální měření, při kterém zjišťovali na jak velké vzdálenosti jsou zařízení schopna komunikovat a jak velké množství chyb při komunikaci vzniká. Měření prováděli ve dvou rozdílných prostředích v Singapuru, kdy přenášeli data o velikosti 10 bajtů přes několik koncových uzlů. Testování pro prostředí s přímou viditelností mezi zařízeními probíhalo u pláže na vzdálenosti 9,08 km a pro prostředí s nepřímou viditelností v kampusu Nanyang Technological University na ploše 3×3 km, kde výhled blokovaly výškové budovy (vysoké v průměru ≥ 100 m). Výsledky měření, při kterém se měnilo nastavení SF, jsou vyobrazeny v grafech na obrázku 2.7 [12].



Obrázek 2.7: Grafy poměru přijatých a nepřijatých paketů v závislosti na vzdálenosti při různých nastavení SF. Vlevo s přímou viditelností, vpravo bez přímé viditelnosti mezi zařízeními [12].

K této práci jsem využil modul Ra-02 od firmy Ai-Thinker. Tento modul je napájen 3,3 V, vysílá na frekvenci 433 MHz s maximálním vysílacím výkonem 18 ± 1 dBm a ke komunikaci s mikrokontrolérem využívá komunikační linky SPI [13].

2.2.2 Mikrokontrolér

Mikrokontroléry známé také jako jednočipové mikropočítače jsou v podstatě spojení všech důležitých částí mikropočítače do jednoho čipu. Mezi hlavní části patří řadič, aritmetická jednotka (liší se dle používané délky slova 4, 8, 16 nebo 32 bitů), paměť programu (kódová paměť), paměť dat a periferní obvody pro vstup a výstup dat. Mimo hlavní části obvykle mikrokontroléry obsahují také generátory hodinového signálu, obvody pro kontrolu správné činnosti mikrokontroléru, obvody pro programování kódové paměti, A/D a D/A převodníky, řadiče přerušení a jiné [14].

V této práci jsem použil 8-bitový AVR mikrokontrolér ATmega328P, který je používán v některých deskách Arduino. Tento mikrokontrolér může být napájen napětím od 1,8 V do 5,5 V. Při použití napájení o napětí 3,3 V je limitována frekvence externího oscilátoru do 10 MHz [15]. Například vývojová deska Arduino Pro Mini existuje ve dvou verzích, jedna pracuje s napětím 3,3 V a oscilátorem o frekvenci 8 MHz, druhá verze pracuje s napětím 5 V a oscilátorem o frekvenci 16 MHz [16].

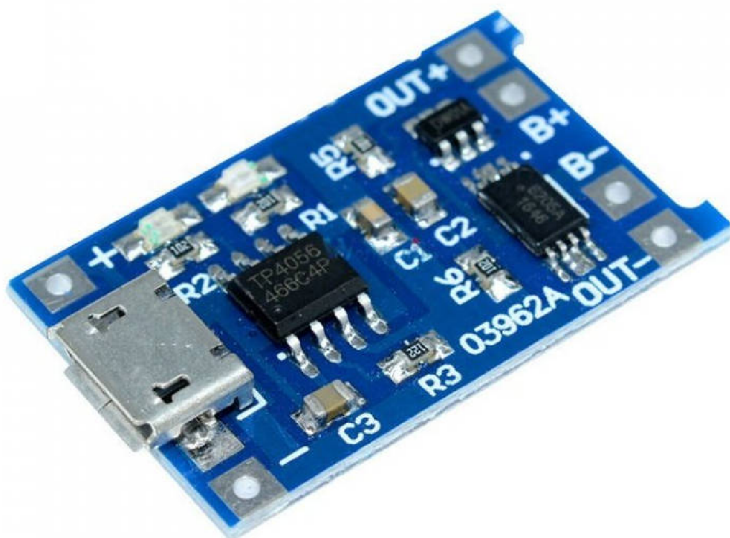
Pro nahrávání kódu do desky Arduino Pro Mini a komunikaci s PC lze využít programátor FTDI Basic Breakout, který využívá integrovaný obvod FT232R [16, 17]. Tento čip slouží pro převádění sériové komunikace UART o úrovních logických napětí 5 V, 3,3 V, 2,8 V a 1,8 V na USB 2.0. FT232R má integrovanou 1024 bitovou EEPROM paměť sloužící pro uchování konfigurace čipu, která lze změnit pomocí programu FT_PROG přes USB port. V paměti EEPROM je uložena např. konfigurace režimu přenosu nebo konfigurace I/O pinů CBUS [18].

2.2.3 Napájení a nabíjení

Návrh napájení je důležitou součástí návrhu, jelikož je důležité aby všechny komponenty byly napájeny dostatečným proudem a správným napětím, které je v čase konstantní. Výkonové zdroje napájení, kde se proudový odběr pohybuje od 0,1 A do 1 A, výjimečně až 5 A, bývají realizovány jako sériové stabilizátory. Kvalitní sériové stabilizátory se vyrábějí v podobě integrovaných obvodů a ke svojí funkci využívají princip operačního zesilovače [19]. Mezi běžně používané patří např. lineární LDO (Low Dropout) stabilizátor AMS1117, vyráběný s nastavitelným nebo s fixním napětím 1,5 V, 1,8 V, 2,5 V, 2,85 V, 3,3 V a 5 V. AMS1117 dokáže pracovat s rozdílem napětí od 1 V mezi vstupním a výstupním napětím

a dodávat proud až 1 A. Regulátor má integrované ochrany proti zkratu a tepelnému přetížení, která při překročení teploty 165 °C regulátor odpojí od napájení [20].

V dnešní době nejrozšířenějším zdrojem energie v přenosných zařízeních (např. mobilní telefony) jsou nabíjecí lithiové baterie. Pro nabíjení lithiových baterií lze sehnat různé moduly využívající metodu nabíjení CC/CV (Constant Current / Constant Voltage), jejichž cena se pohybuje do 20 Kč. Některé tyto moduly obsahují kromě integrovaného obvodu pro nabíjení, kterým je u těchto modulů často TP4056, také ochrany proti zkratu, nadproudu či podbití baterie. Nabíjecí proud u modulů s integrovaným obvodem TP4056 lze nakonfigurovat výměnou rezistoru R_{prog} . Jak takový nabíjecí modul s ochranou vypadá je vidět na obrázku 2.8 [21, 22].



Obrázek 2.8: Nabíječka lithiových akumulátorů TP4056 s ochranou a portem microUSB [21]

2.3 Python

Python je programovací jazyk pro univerzální použití podobný jiným programovacím jazykům jako jsou C++, JavaScript, C# od společnosti Microsoft nebo Java od společnosti Oracle. Mezi přední vlastnosti patří jeho funkčnost na většině operačních systémech, včetně Windows, MacOS a Linux. Taktéž je výhodou velké množství knihoven, které rozšiřují základní funkce jazyka. V roce 1980 tento jazyk navrhl Guido van Rossum v Centrum Wiskunde & Informatica v Nizozemsku. Pojmenován byl podle jednoho z Guidových oblíbených pořadů „Monty Pythons Flying Circus“ [23]. V nynější době existují dvě verze jazyka Python:

- Python 2 spuštěný v říjnu 2000 a v dnešní době stále používaný.
- Python 3 spuštěný v prosinci 2008, který je významnou revizí a není zpětně kompatibilní. [23]

K vytvoření programu pro zadávání sekvencí zážehu jsem využil verzi Python 3.10.

2.3.1 pySerial

Pro přístup k sériovým portům počítače, na kterém bude program spuštěn, slouží knihovna pySerial. Jedná se o knihovnu poskytující backendy pro Python 2.7 nebo 3.4 a novější, fungující na operačních systémech Windows, OSX, Linux, BSD (nejspíše všechny operační systémy kontabilní s POSIX) a IronPython. Modul nazvaný „serial“ automaticky vybírá příslušný backend pro daný operační systém [24]. Přesto je nutné při programování dbát pozornost na rozdílnost mezi operačními systémy, např. rozdílné pojmenování sériových portů na OS Windows a Linux. U Windows bývají sériové porty označovány např. „COM0“, kdežto u Linux jsou označovány např. „/dev/ttyUSB0“.

Po otevření portu sériové komunikace nám knihovna umožňuje ze sériové linky data číst, ale také i data zapisovat. Pro úspěšné otevření portu je zapotřebí správně nadefinovat jaký port se má otevřít a s jakými parametry. Všechny parametry jsou v knihovně již předdefinovány a pokud by nám výchozí nastavení vyhovovalo nemusíme je již definovat znova, stačí pouze zadat port a sériovou linku otevřít. Jaké porty máme k dispozici zjistíme pomocí nástroje „serial.tools.list_ports“, který je v knihovně obsažen [24].

2.3.2 Tkinter

Knihovna Tkinter poskytuje přístup k velmi bohaté sadě grafických nástrojů zvané Tk. Tk jsou multiplatformní sada nástrojů standardního GUI pro řadu programovacích jazyků včetně Pythonu, Tcl a Perlu. Poskytuje nám všechny běžně používané komponenty (obvykle nazývané widgety), jsou to např. tlačítka, vstupní pole, zaškrtačací políčka a další [25].

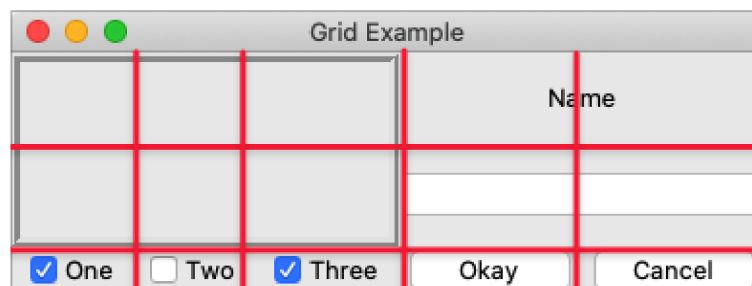
Widgety se zde nejčastěji umísťují pomocí mřížky do buněk, jako například data v tabulkových editorech. Každá buňka může obsahovat pouze jeden widget a má svůj index, který je složen z indexu sloupce a řádku. Na obrázku 2.9 můžeme vidět indexování buněk začínající nulou z počátku v levém horním rohu odkud se po jedné rozšiřuje dál. Takové indexování není nutné dodržovat. Je možné v indexování přeskokovat a vytvářet tak volná místa, které nebudou rozpořeny nijak ovlivňovat a je možné je využít pro pozdější přidávání widgetů. Například můžeme mít mřížku s indexy sloupců 1, 2, 10, 11 a 12. Šířka sloupce závisí na šířkách widgetů, které sloupec obsahuje, stejně tak závisí výška řádku

		Columns		
Rows	(0,0)	(1,0)	(2,0)	
	(0,1)	(1,1)	(2,1)	
	(0,2)	(1,2)	(2,2)	
	(0,3)	(1,3)	(2,3)	

(Column, row)

Obrázek 2.9: Indexování buněk v mřížce pro umístování widgetů [26]

na výškách widgetů obsažených v řádku. Další užitečné funkce, které stojí za zmínku, jsou např. slučování buněk nebo centrování widgetů v buňce [26]. Na obrázku 2.10 můžeme vidět jak může vypadat uživatelské rozhraní s rozmístěním widgetů do mřížky. Pro zvýraznění je mřížka v obrázku vyznačena červenými čarami.

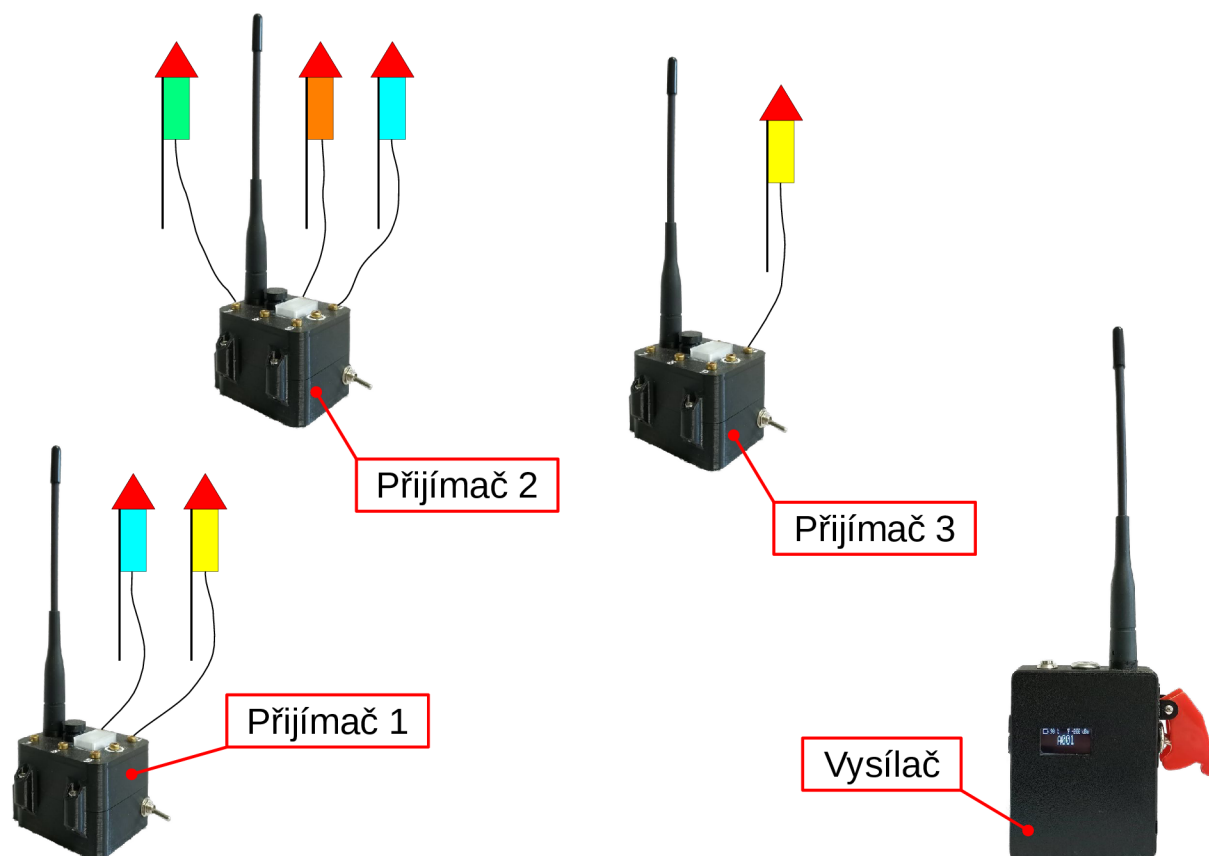


Obrázek 2.10: Příklad uživatelského rozhraní s widgety umístěných do mřížky [27]

3 Postup a výsledky řešení

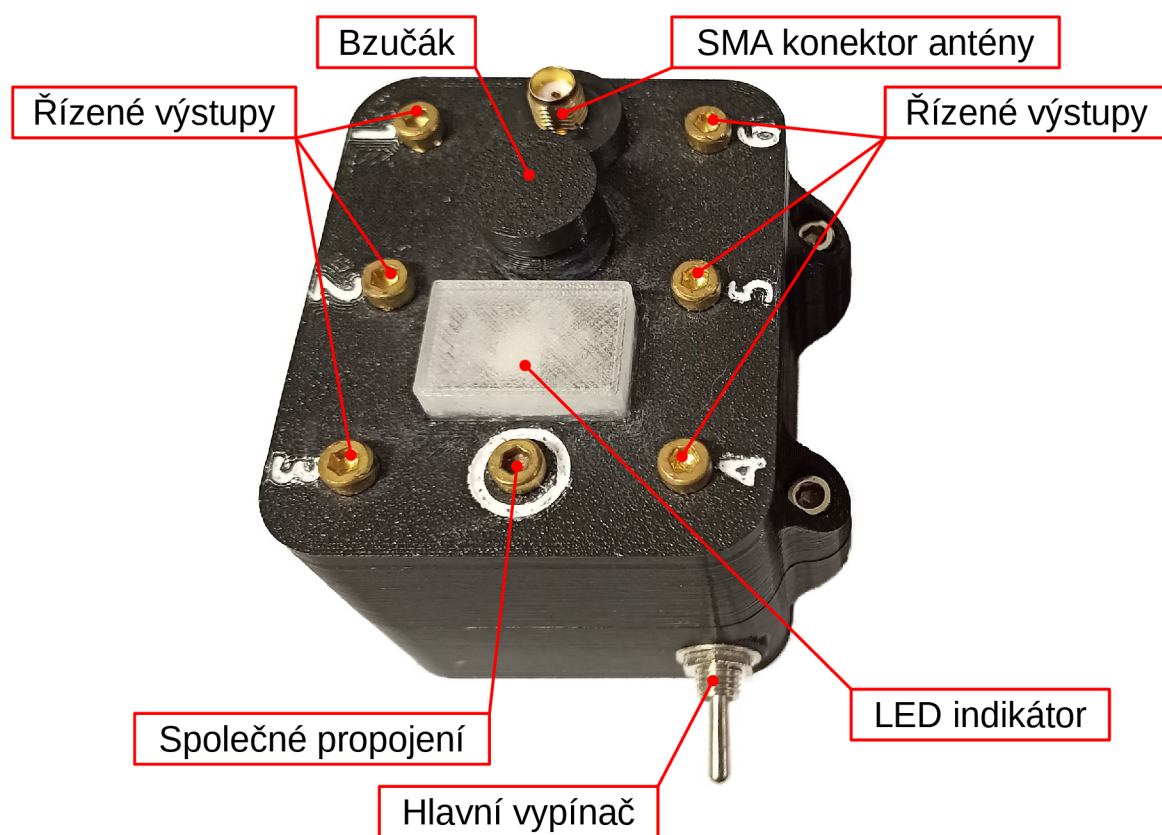
3.1 Popis funkce systému

Mnou navržený dálkový systém pro zážeh zábavní pyrotechniky se skládá z vysílače a vícero přijímačů. Na každém přijímači se nachází šest výstupů, které mohou být aktivovány současně, jednotlivě podle nastavených sekvencí nebo postupně potvrzením aktivace jednotlivých výstupů pomocí vysílače. Popis částí přijímače se nachází na obrázku 3.2. Ovládání pomocí vysílače (popsaného na obrázku 3.3) je navrženo tak aby bylo pro uživatele co nejbezpečnější a nejjednodušší. Na obrázku 3.1 je ilustračně vyobrazen vysílač se třemi přijímači, kde u prvního přijímače jsou připojeny dvě rakety, u druhého tři rakety a u třetího jedna raketa.



Obrázek 3.1: Schéma dálkového systému pro zážeh zábavní pyrotechniky

V současné verzi je vysílač schopen vyslat příkaz k zážehu 255 výstupům. Tento počet je omezen pouze programově a je ho případně možné upravit v kódu vysílače. Počet použitých přijímacích jednotek může být v podstatě neomezený.



Obrázek 3.2: Popis přijímače dálkového systému pro zážeh zábavní pyrotechniky



Obrázek 3.3: Popis vysílače dálkového systému pro zážeh zábavní pyrotechniky

3.1.1 Požadavky na systém

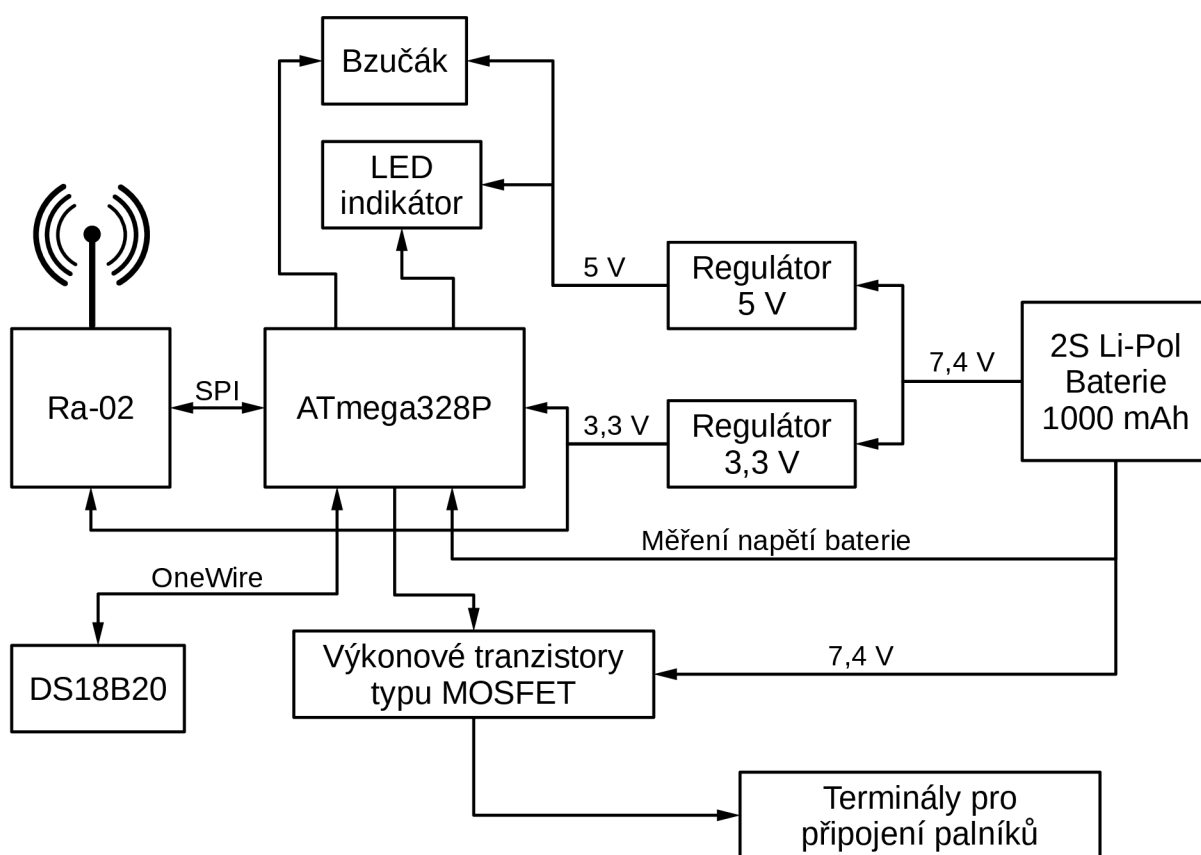
- **Ochrana proti náhodnému zážehu**
Je velmi důležité aby u systému operujícím se zábavní pyrotechnikou nemohlo dojít k zážehu přijetím náhodného signálu.
- **Signalizace zážehu zvukovou a světelnou signalizací**
Zážeh každého palníku musí být signalizován zvukovou a světelnou signalizací.
- **Spolehlivost systému**
Systém by měl být navržen tak aby se zamezilo chyb při zpracování a přenosu dat.
- **Bezdrátová komunikace na vzdálenost 100 metrů**
Bezdrátový přenos by měl spolehlivě fungovat na vzdálenost minimálně 100 metrů, tak aby bylo možné dodržet bezpečnou vzdálenost od zábavní pyrotechniky.
- **Hodina pohotovostního provozu na baterii**
Systém by měl vydržet v pohotovostním režimu, tedy v režimu kdy je systém zapnutý a čeká na vstup od uživatele, napájený z baterií alespoň po dobu jedné hodiny. Během této doby by měl být schopen systém zážehu připojených palníků.
- **Uživatelská přívětivost systému**
Systém by měl být pro uživatele jednoduše ovladatelný a zároveň musí obsahovat bezpečnostní prvky zabráňující náhodnému odeslání příkazu k zážehu.

3.2 Sestavení prototypu

3.2.1 Návrh DPS

Na návrhy DPS jsem využil počítačového programu KiCad. Pro vytvoření DPS je nejprve zapotřebí vytvořit schéma zapojení, kde se zároveň k jednotlivým součástkám přiřazuje jejich pouzdro.

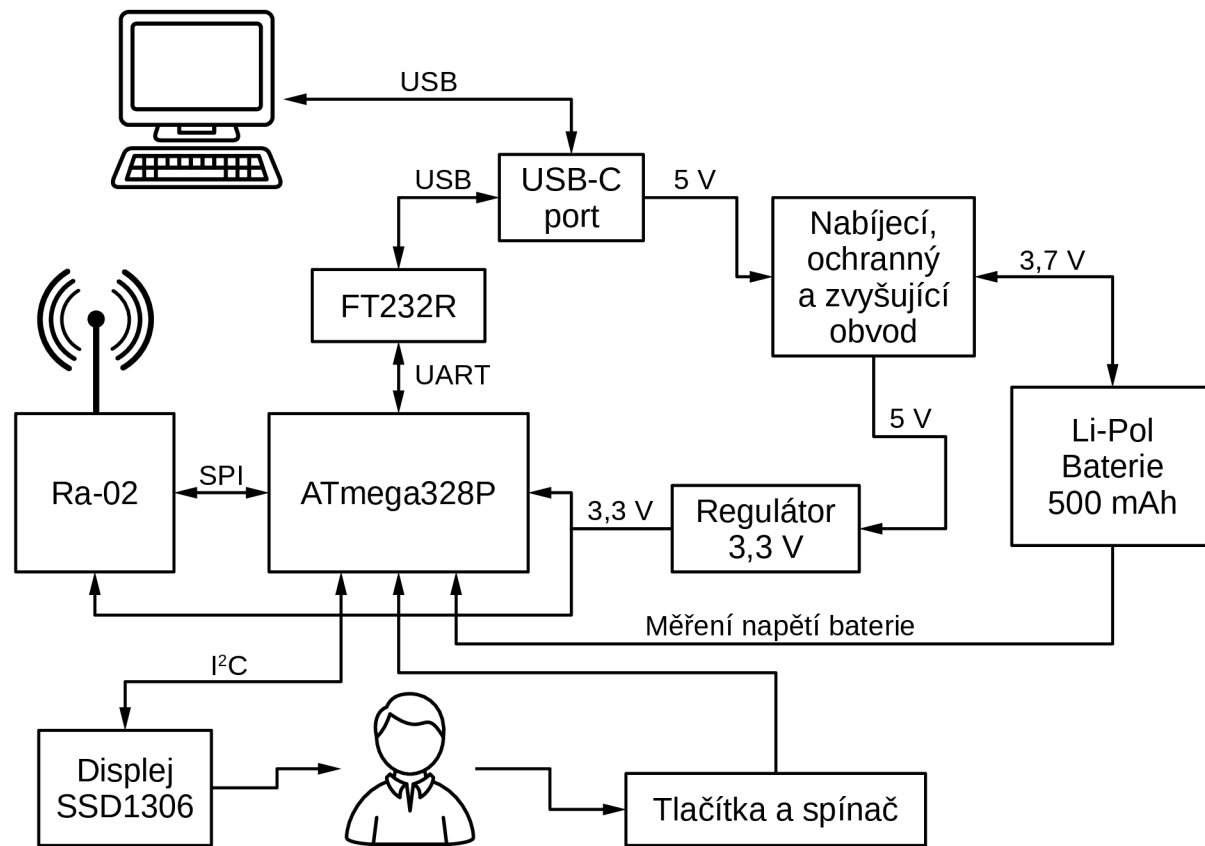
Při tvorbě schématu zapojení vysílače a přijímače jsem vycházel z ukázkového schématu výrobce modulu Ra-02 a schématu zapojení vývojové desky Arduino Pro Micro. Jelikož má tyto součástky vysílač i přijímač stejné, stejná je i tato část návrhu ve schématu. Z důvodu využití modulu Ra-02, který využívá pro napájení a komunikaci napětí 3,3 V, jsem se rozhodl celý obvod včetně mikrokontroléru napájet tímto napětím a pro regulaci napětí na tuto úroveň jsem využil LDO regulátor AMS1117 zmíněný v kapitole 2.2.3.



Obrázek 3.4: Blokové schéma zapojení přijímače

Jako převodník sériové komunikace na USB jsem využil FT232R zmíněný v kapitole 2.2.2. V případě vysílačky je čip umístěn přímo na DPS, jelikož je nutné aby vysílačka mohla komunikovat s PC. Ovšem u přijímacích jednotek by tento čip sloužil pouze k nahrávání kódu, což uživatel při běžném používání nevyužije a proto DPS přijímací jednotky tento čip neobsahuje. V případě nahrávání kódu do přijímače se k desce připojí externí programátor s tímto čipem a po nahrání kódu je programátor odpojen. Tímto způsobem se ušetřilo místo na DPS a také cena přijímací jednotky. Pro návrh zapojení převodníku k mikrokontroléru jsem využil schéma z dokumentace čipu FT232R. K indikaci komunikace jsem do zapojení přidal dvě oranžové LED, které jsem taktéž zapojil dle dokumentace. Na obrázku 3.7 se jedná o LED označené D1 (indikující vysílání dat) a D2

(indikující příjem dat) [18].

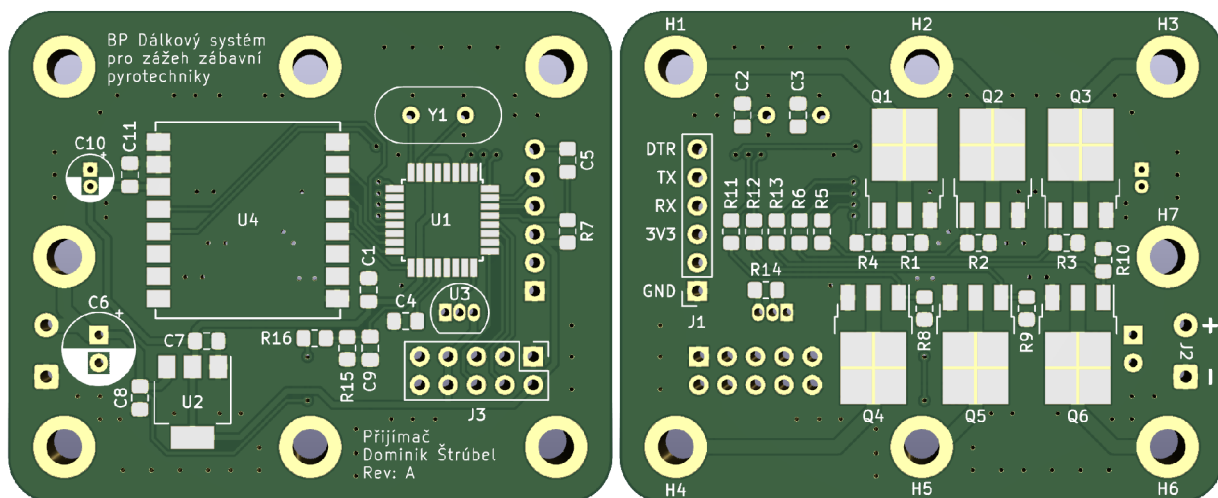


Obrázek 3.5: Blokové schéma zapojení vysílače

Z důvodu nutnosti vytvoření propojení mezi vysílačem a PC pro komunikaci, realizovaného pomocí USB portu, jsem využil tento USB port také k nabíjení vysílače. Pro nabíjení a ochranu LiPol baterie jsem využil obvod vytvořený YouTube tvůrcem vystupujícím pod jménem „GreatScott!“ a obvod mírně upravil pro aplikaci v mé práci. Obvod obsahuje ochranu proti přílišnému vybití nebo nabití baterie, ochranu proti zkratu, nabíjecí obvod a obvod pro zvyšování výstupního napětí na 5 V, který je schopen toto napětí dodávat i při zátěži 1 A. Pro nabíjení metodou CC/CV je v obvodu využito integrovaného obvodu TP4056. Nastavení nabíjecího proudu se zde provádí připojením rezistoru k pinu integrovaného obvodu označeném „PROG“. Na obrázku DPS 3.7 je tento rezistor označen R12. Hodnotu nabíjecího proudu pro Li-Pol baterii GeB 503035 o kapacitě 500 mAh, použitou ve vysílači, jsem zvolil přibližně 200 mA. Tato hodnota nabíjecího proudu dle dokumentace integrovaného obvodu TP4065 odpovídá rezistoru o odporu 5,8 k Ω . K indikaci stavu baterie jsou zde použity dvě LED, červené barvy označené D4 a zelené barvy označené D5 na obrázku 3.7. Svítící červená LED značí nabíjení baterie a svítící zelená LED plně nabitou baterii. Příklad, kdy svítí zelená LED a červená LED bliká, značí problém v připojení baterie [28, 29].

Pro spínání výstupů H1–H6 u přijímače jsem použil tranzistory typu MOSFET s N kanálem IRLR3103 označené Q1–Q6. Vývod H7 slouží jako společné propojení, připojené ke kladnému pólu baterie. Označení výstupů a tranzistorů lze vidět na obrázku 3.6. Použité tranzistory jsou schopny pracovat při napětí až 30 V mezi source a drain, maximální dlouhodobý proud protékající výstupem drain je 55 A, nicméně tento proud je omezen

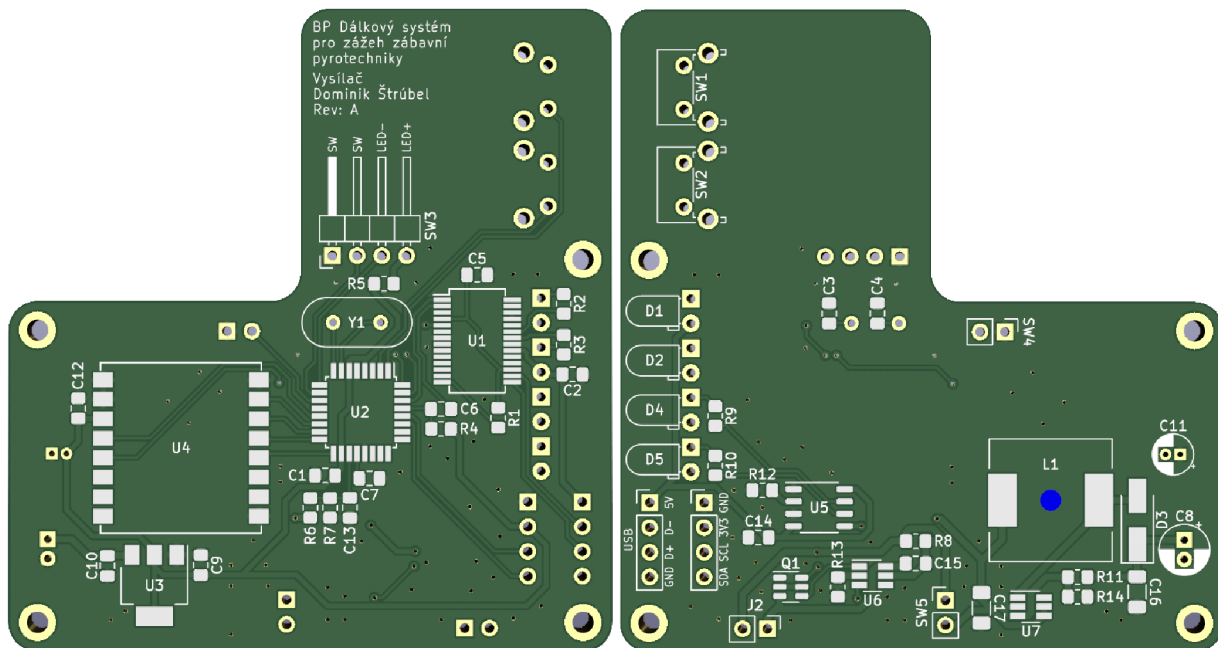
maximálním proudem pouzdra na 20 A [30].



Obrázek 3.6: Náhled DPS přijímače vytvořený programem KiCad

Po dokončení schématu i s přiřazenými pouzdry součástek jsem přikročil k návrhu rozložení součástek na desce. V tomto kroku bylo důležité aby byly součástky rozloženy tak aby se různé vývody, tlačítka nebo LED indikátory nacházely na místech kde jsou na desce potřeba. Součástky by taktéž měly být rozloženy s co nejlepším využitím místa a propojení mezi součástkami by mělo být co nejkratší aby se minimalizovali parazitní jevy.

Do návrhu přijímače jsem také zakomponoval digitální teploměr DS18B20, pro monitorování teploty uvnitř přijímače. Teploměr se na DPS nachází, pod označením U3, blízko mikrokontroléru, jak je vidět na obrázku 3.6.

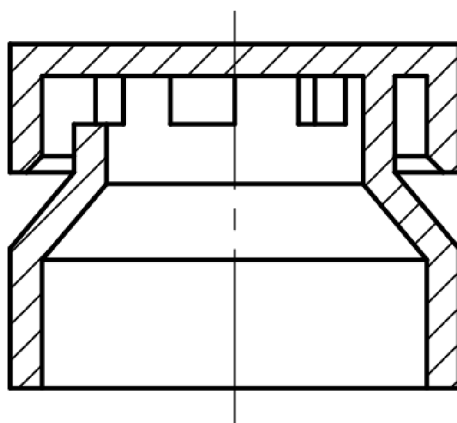


Obrázek 3.7: Náhled DPS vysílače vytvořený programem KiCad

3.2.2 Výroba krabiček

Pro návrh krabiček jsem zvolil 3D CAD program SolidWorks 2020 a jako předlohu jsem využil 3D modely DPS exportované z programu KiCad do formátu STEP, který je univerzálně použitelný ve většině programech pro tvorbu a práci s 3D modely. Za využití těchto 3D modelů DPS jsem mohl navrhnout krabičky přesně na míru, což značně usnadnilo práci a minimalizovalo množství zkušebních výtisků, na kterých by bylo potřeba testovat správné usazení všech dílů.

Návrh obou modelů jsem se snažil dělat co nejvíce odolné proti vniknutí prachu a vlhkosti z důvodu, že by tyto nepříznivé faktory mohly negativně ovlivnit funkčnost zařízení. Jelikož jsou krabičky dvoudílné jsou spoje utěsněny pomocí těsnění vytisknutých z flexibilního materiálu. Na obrázku 3.8 je vyobrazena krytka bzučáku v řezu, která je navržena tak aby bylo minimalizováno vniknutí vody v případě deště, jelikož díry v horní části krytky jsou překryty převisem a brání tak vniknutí vody dopadající na horní část krytky.



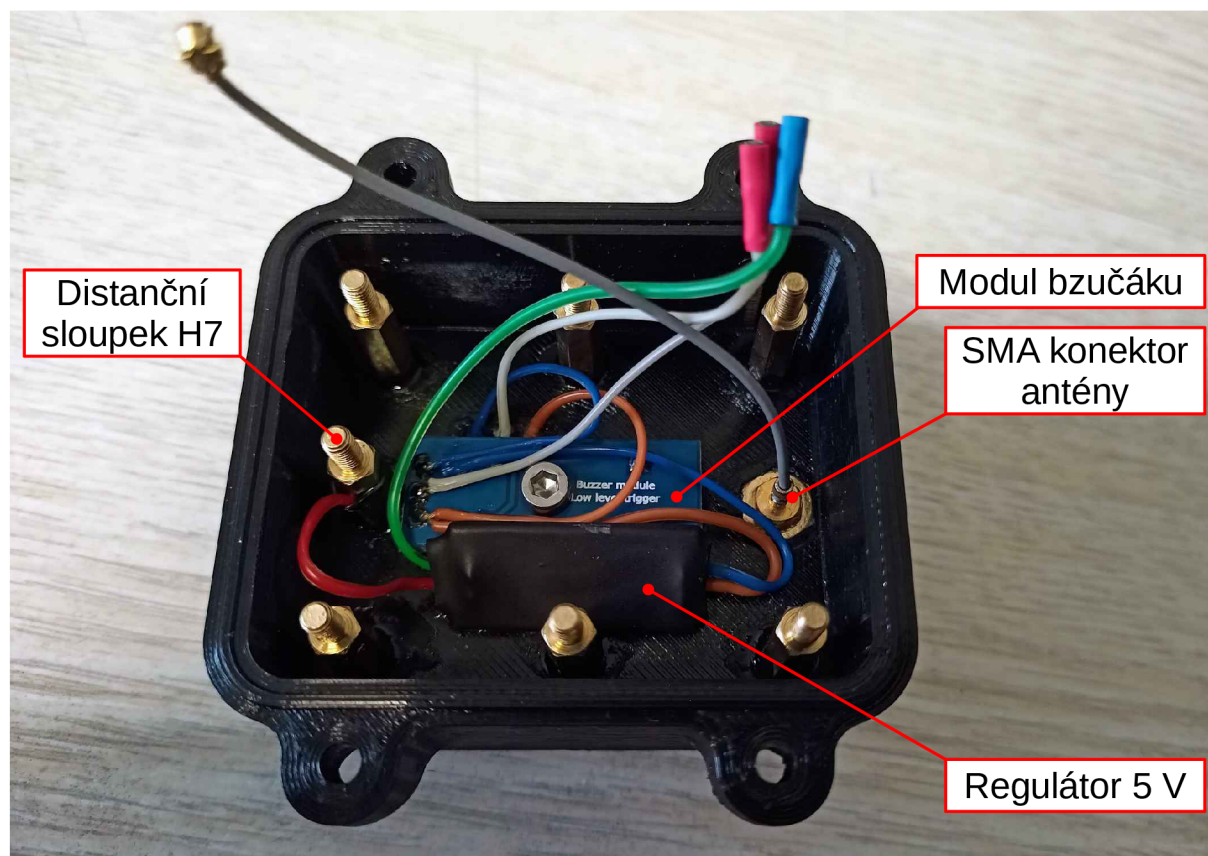
Obrázek 3.8: Řez modelem krytky bzučáku

Po vytvoření počítačových 3D modelů jsem pomocí PrusaSliceru vytvořil G-kód pro tiskárnu a modely vytiskl. K tisku pevných dílů jsem použil materiál PETG a pro flexibilní díly materiál TPE 88 (RubberJet Flex).

3.2.3 Kompletace řešení

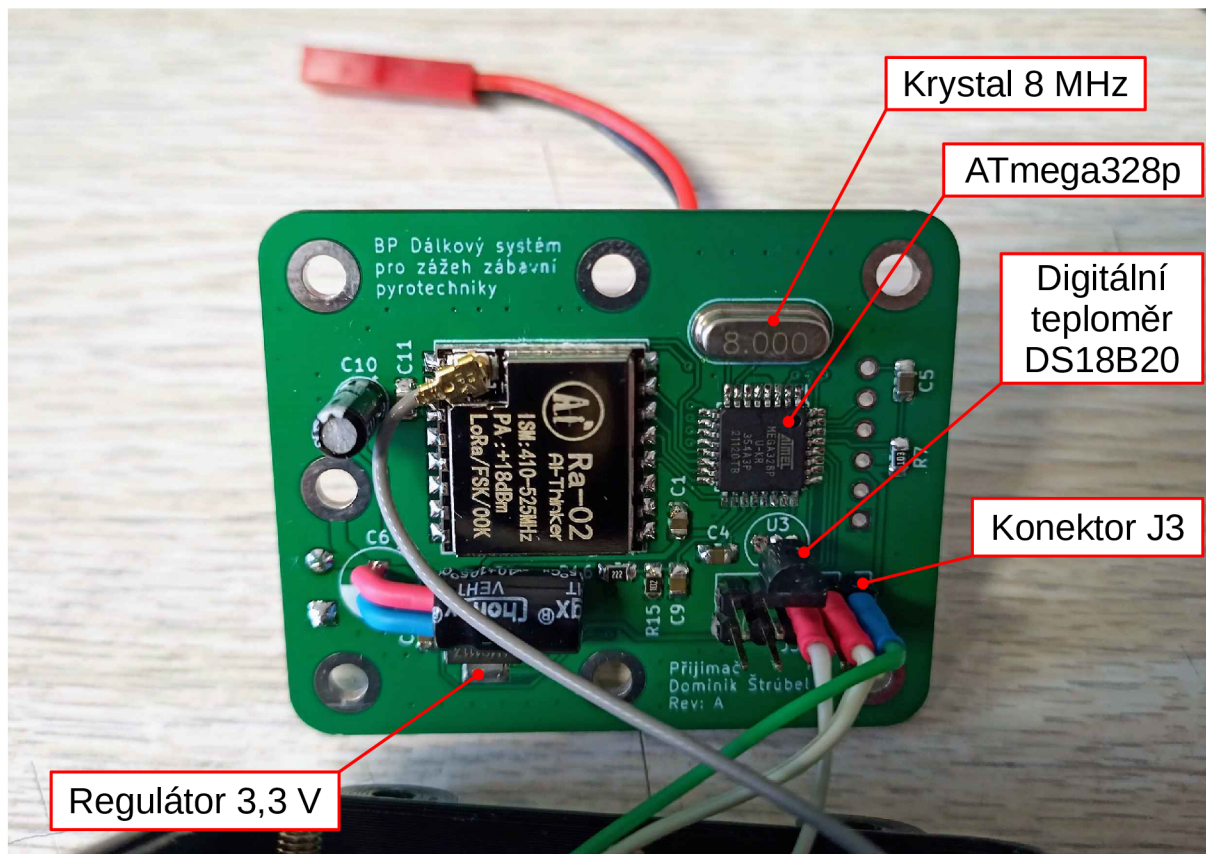
V přijímači je DPS uchycena na distančních sloupcích (obrázek 3.9) pomocí mosazných matic a měděných pérových podložek přes pokovené díry H1–H7 (obrázek 3.11). Distanční sloupky zároveň procházejí stěnou krabičky a slouží jako šroubovací svorkovnice pro připojení palníků. Chtěl jsem se takto vyhnout propojování DPS se svorkovnicí vodiči a využití distančních sloupků se jevila jako dobrá alternativa, jelikož jsou využity k vedení elektrického proudu, k uchycení DPS a zároveň jako svorkovnice. Pro připojení baterie jsou zde na DPS vývody J2, na kterých je připájen konektor typu JST používaný na bateriích pro RC modely. Tento konektor svým tvarem zamezuje prohození polarity při připojování baterie a je schopný přenášet vysoký proud. Vodič mezi tímto konektorem a DPS je přerušen hlavním vypínačem na spodní části krabičky. Světelný LED indikátor je realizován pomocí jednoho segmentu adresovatelného LED pásku WS2812B přilepeného k venkovní straně krabičky zakrytého krytkou z průhledného plastu utěsněnou lepidlem, tak aby se zabránilo vniknutí vody. Pro akustickou indikaci jsem využil modulu bzučáku, přichyceného šroubem z vnitřní strany krabičky, který prochází dírou v krabičce a z venkovní

strany krabičky je zakryt krytkou bzučáku z obrázku 3.8. Krytka bzučáku je taktéž, proti vniknutí vody, utěsněna lepidlem. Indikátory jsou k DPS připojeny pomocí konektoru J3 a jsou napájeny 5 V stabilizátorem AMS1117, napájeným přímo z baterie přes distanční sloupek H7. Připojení indikátorů ke konektoru na DPS lze vidět na obrázku 3.10. K napájení přijímačů jsem použil dvoučláňkové Li-Pol baterie s nominálním napětím 7,4 V a kapacitě 1000 mAh. Baterie se vkládá do prostoru spodní části krabičky. Při uzavírání je mezi DPS a baterií vložen molitan sloužící pro vymezení prostoru. Nabíjení baterie se provádí pomocí externí nabíječky s balancérem.

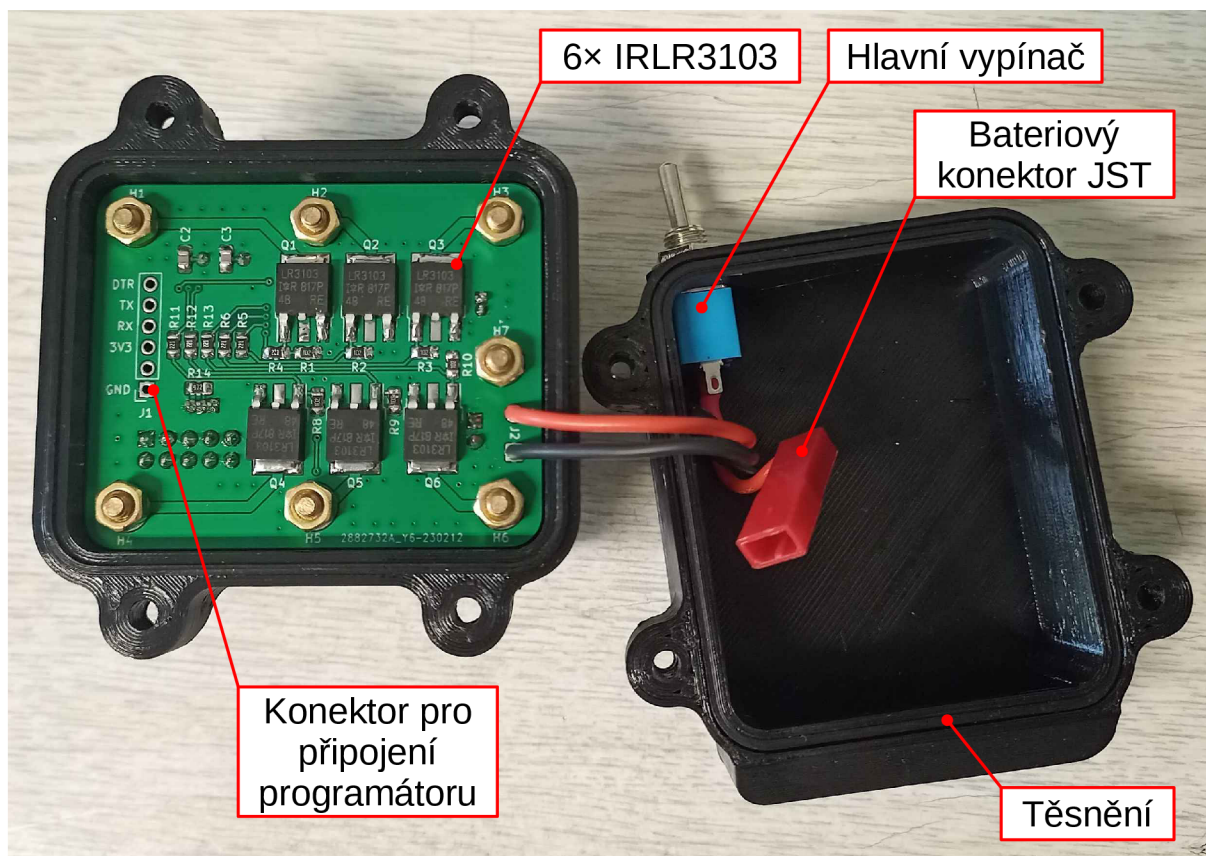


Obrázek 3.9: Zapojení součástí v horní části krabičky přijímací jednotky

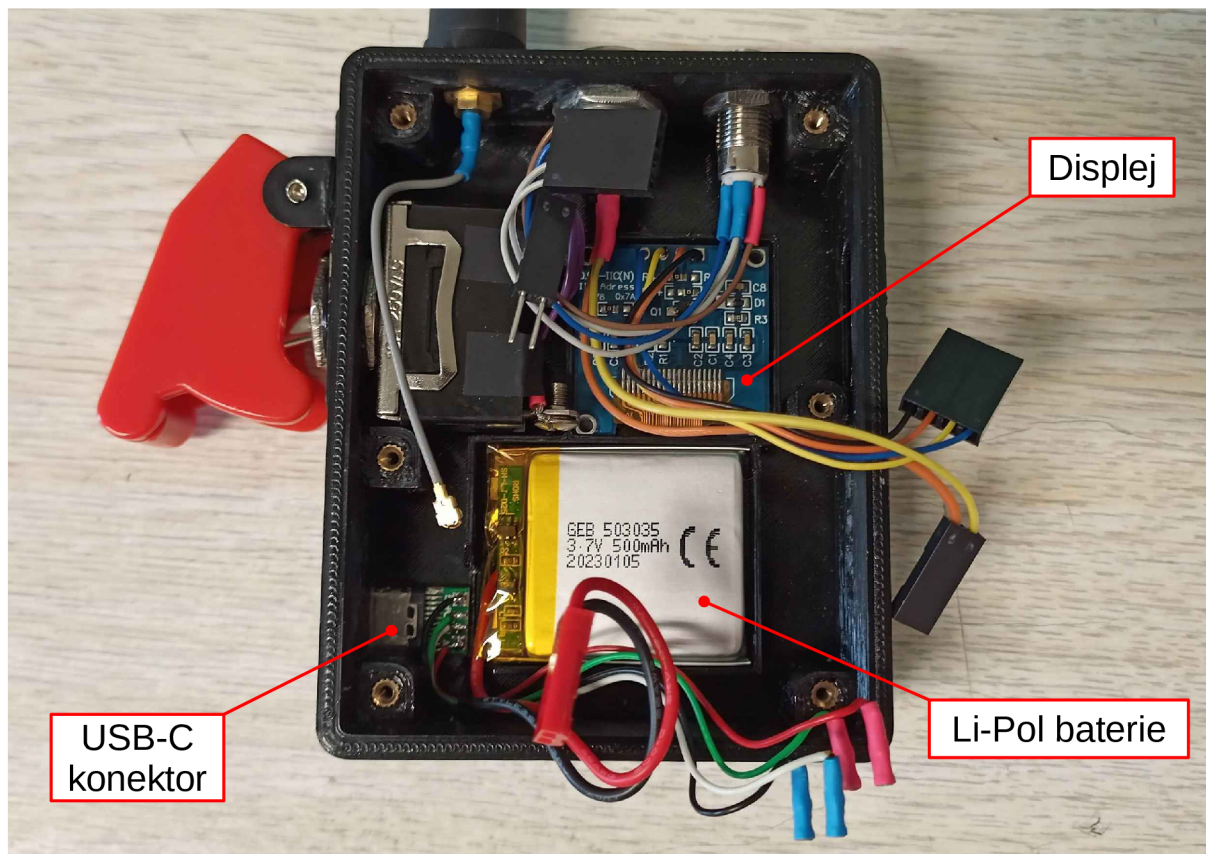
DPS ve vysílači se přichycuje ke krabičce spolu s víkem krabičky. Šrouby přichycující víko krabičky procházejí skrze DPS (obrázek 3.14) a distanční sloupky vytištěné z flex materiálu umístěné na šroubu mezi DPS a víkem při zašroubování víka přitlačí DPS ke krabičce a zamezují tak jeho pohybu. Baterie je vložena do přihrádky v krabičce vysílače (obrázek 3.12) a proti pohybu je zajištěna vloženým molitanem mezi baterkou a DPS. Port USB-C je proti vniknutí nečistot a vlhkosti chráněn krytkou z Flex materiálu. Indikační LED na obrázku 3.13 jsou mezi sebou rozděleny černou plastovou krytkou aby se tak zamezilo prosvítání jednotlivých LED mezi sebou. Pro zobrazení informací jsem zde použil jednobarevný OLED I²C displej SSD1306 s rozlišením 128×64 pixelů.



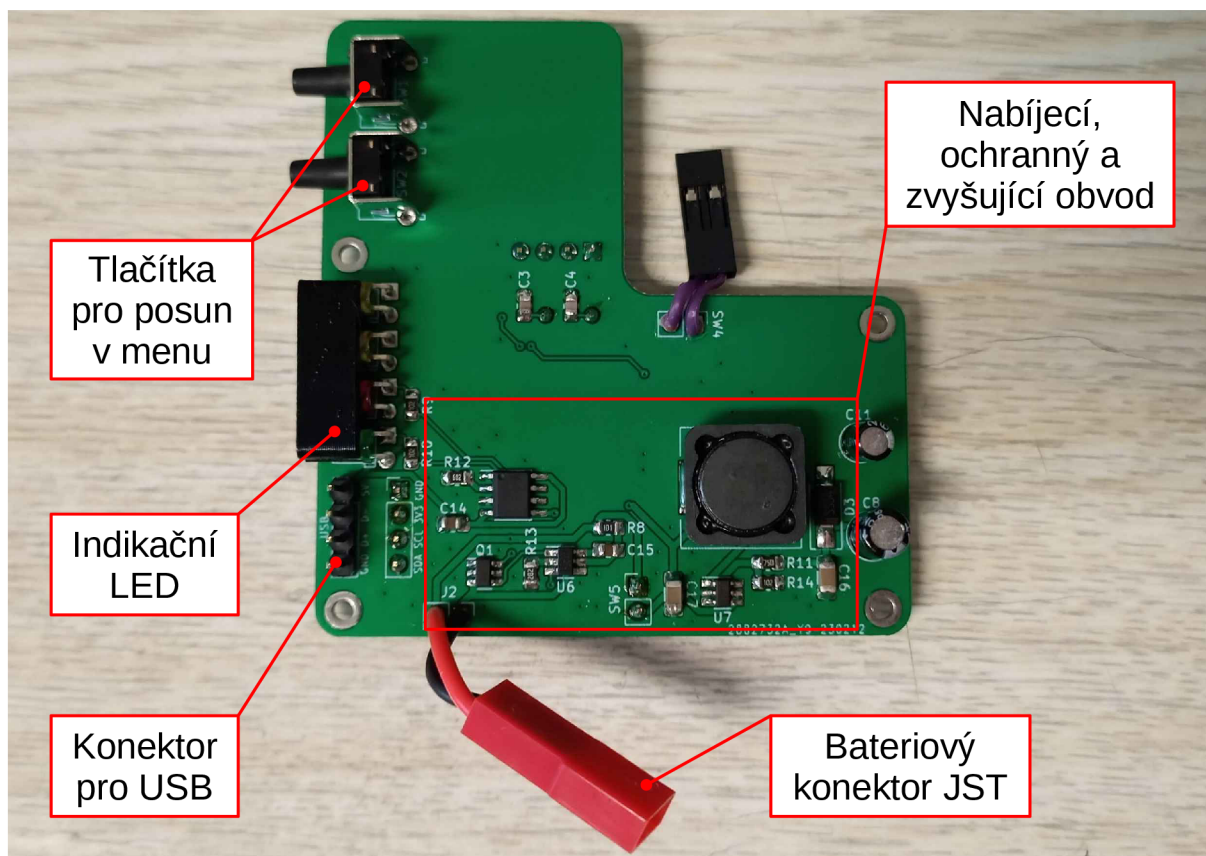
Obrázek 3.10: Propojení horní části krabičky s přední stranou DPS u přijímací jednotky



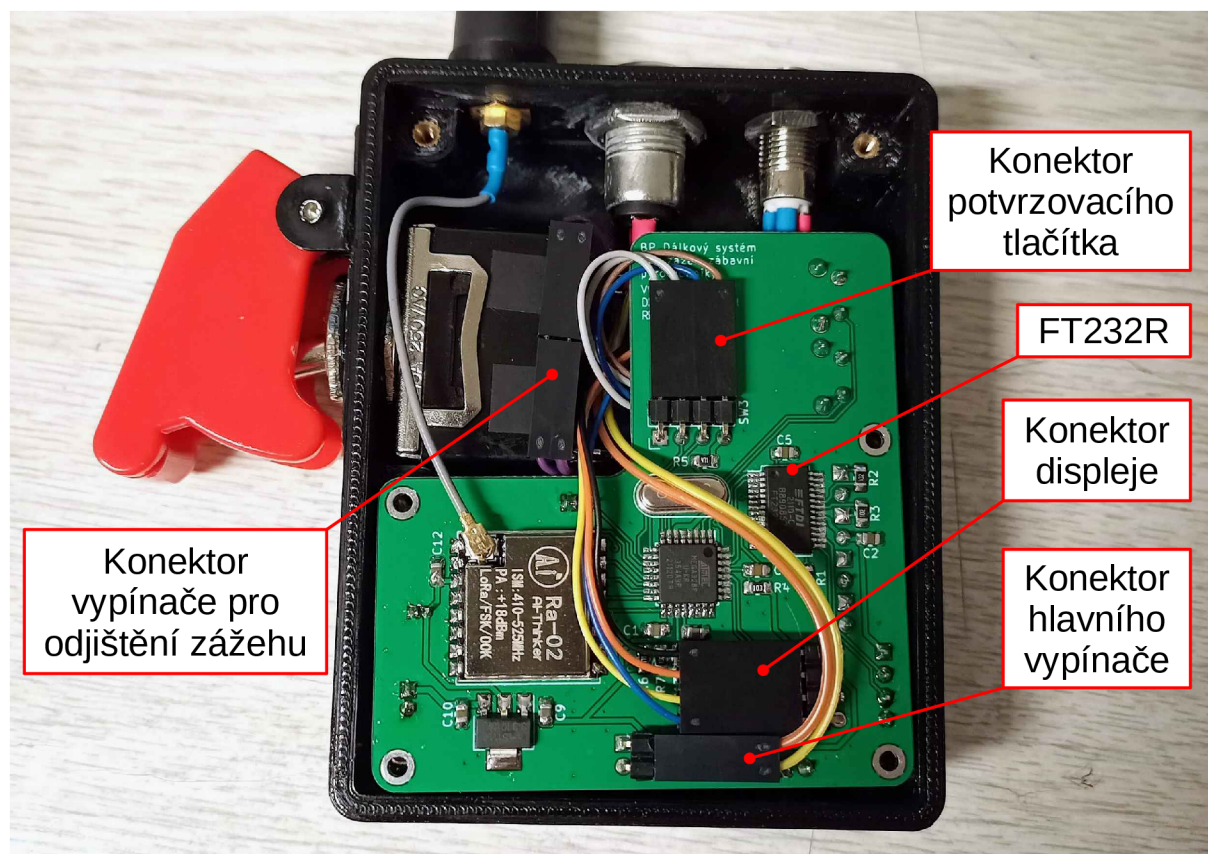
Obrázek 3.11: Náhled do zapojení přijímací jednotky s přichycenou DPS



Obrázek 3.12: Rozmístění komponent pod DPS vysílače



Obrázek 3.13: Zadní strana DPS vysílače



Obrázek 3.14: Náhled do zapojení vysílače a propojení s DPS

3.3 Programování

3.3.1 Firmware pro vysílač a přijímač

K programování mikrokontroléru jsem se rozhodl využít prostředí Arduino IDE. Z toho důvodu je nejprve zapotřebí do mikrokontroléru vypálit zavaděč, aby jej bylo možné programovat pomocí zmíněného převodníku FT232R v kapitole 2.2.2. Pro vypálení zavaděče lze využít několik možností, osobně k tomuto procesu využívám vytvořený přípravek sestavený z patice pro pouzdra TQFP-32, převodníku USBasp a základních součástek potřebných pro funkci mikrokontroléru. Pro vypálení zavaděče do mikrokontroléru tak stačí vložit mikrokontrolér do patice, připojit převodník do portu USB počítače a spustit proces vypalování zavaděče ve vývojovém prostředí Arduino IDE. Pro správné vypálení zavaděče je třeba v prostředí Arduino IDE vybrat v záložce „Nástroje“ vývojovou desku: „Arduino Pro or Pro Mini“, procesor: „ATmega328P (3.3V, 8MHz)“ a programátor: „USBasp“.

Díky velké oblíbenosti platformy Arduino mezi lidmi existuje pro tuto platformu spousta knihoven pro zjednodušení práce se všemi možnými přídatnými moduly. Při programování vysílače a přijímače jsem využil následující knihovny:

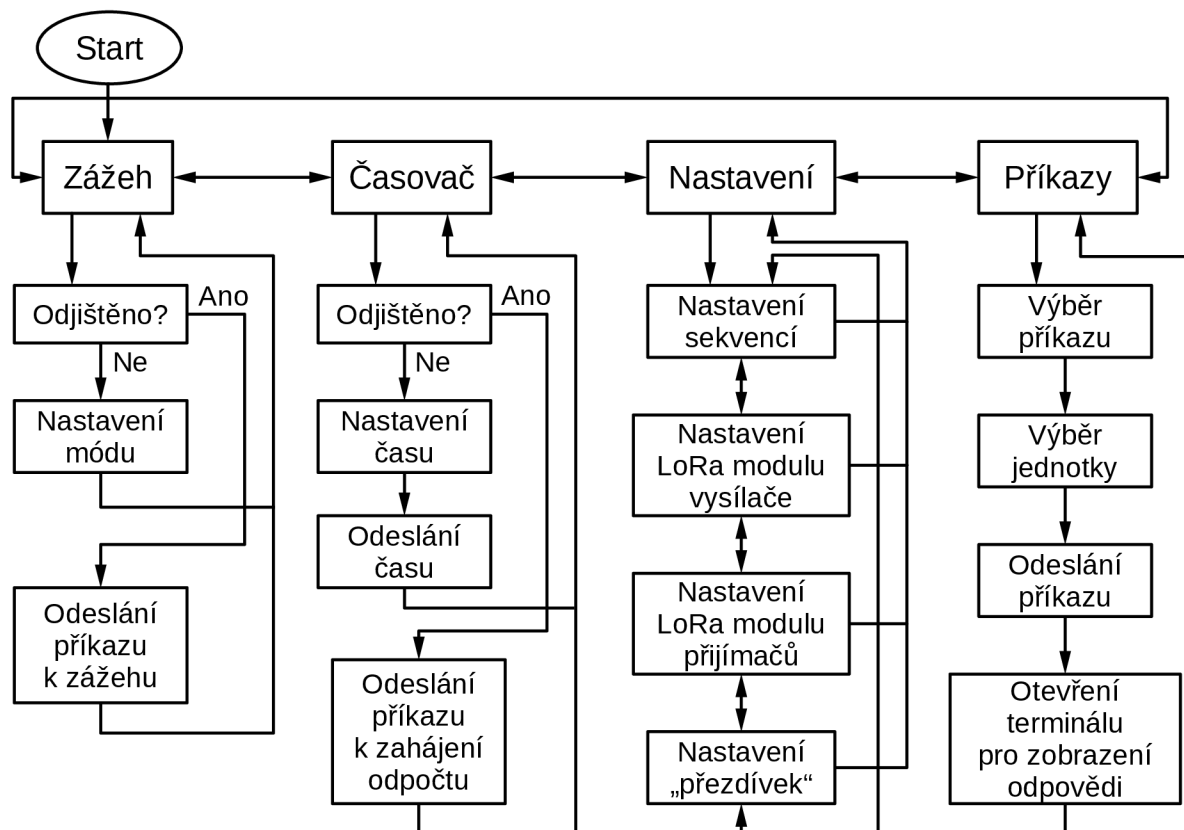
- **Arduino LoRa** - ovládání modulu Ra-02 (nastavování parametrů modulu, vysílání a přijímání dat). Z důvodu využití komunikace SPI mezi mikrokontrolérem a modulem Ra-02 je nutné využít taktéž knihovnu „SPI“ [31]. Knihovna je dostupná z: <https://github.com/sandeepmistry/arduino-LoRa>
- **Adafruit SSD1306** - vykreslování dat na displeji vysílače. Tato knihovna dále ke své funkci využívá knihovny: „Adafruit GFX Library“, „Adafruit BusIO“ a „Wire“.

V případě, že by se jednalo o displej s komunikací SPI místo I²C, byla by místo knihovny „Wire“ použita knihovna „SPI“ [32, 33, 34].

Knihovna je dostupná z: https://github.com/adafruit/Adafruit_SSD1306

- **OneWire** - získávání teploty z digitálního teploměru DS18B20, umístěného na DPS přijímače.
Knihovna je dostupná z: <https://github.com/PaulStoffregen/OneWire>
- **FastLED** - ovládání adresovatelného LED pásku WS2812B.
Knihovna je dostupná z: <https://github.com/FastLED/FastLED>
- **EEPROM** - ukládání nastavení vysílače a přijímače do paměti EEPROM mikrokontroléru.

Knihovny „SPI“, „Wire“ a „EEPROM“ jsou oficiální knihovny Arduino a jsou již předinstalovány v prostředí Arduino IDE [35].



Obrázek 3.15: Blokové schéma funkce programu vysílače

U vysílačky jsem vytvořil menu ve kterém je možné volit mezi různými styly zážehu, nastavovat sekvence zážehu a měnit nastavení vysílače nebo přijímačů. Na obrázku 3.15 je znázorněno blokové schéma popisující toto menu a jeho funkce. V menu se pohybuje pomocí bočních tlačítek na levé straně a pro potvrzení slouží tlačítko na horní straně. Po zapnutí vysílače je vybrána první funkce v menu a na displeji se zobrazí text „A001“. Stisknutím potvrzovacího tlačítka se označí písmeno A a bočními tlačítky je možné přepínat mezi čtyřmi možnostmi:

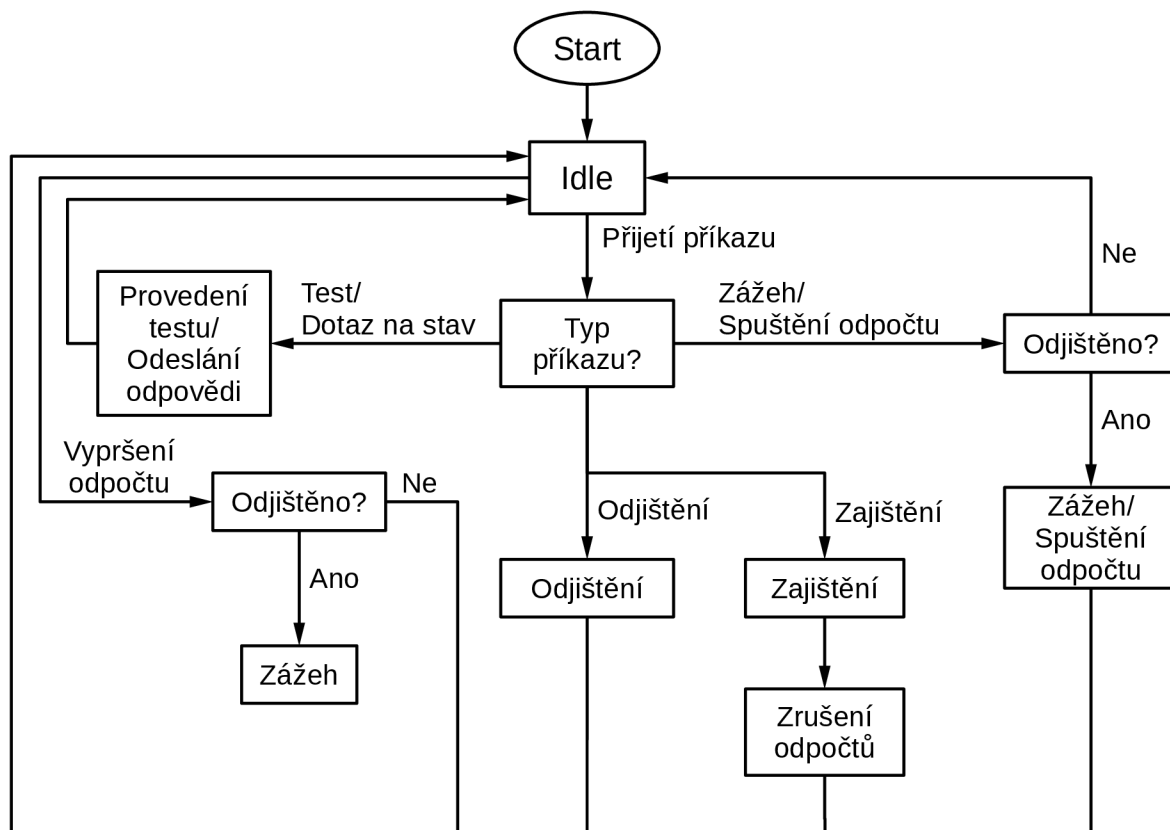
A Reply Required - Odesílání příkazu k zážehu probíhá jednotlivě vzestupně podle čísel jednotlivých výstupů na přijímačích. Každý jednotlivý zážeh je potvrzen potvrzovacím tlačítkem. Vysílač následně vyčkává na odpověď přijímače zda byla zpráva přijata a po přijetí odpovědi pokračuje na další pozici v pořadí.

B NO Reply Required - Stejně jako předchozí možnost A s tím, že vysílač ignoruje odpověď od přijímače a pokračuje na další pozici v pořadí.

C Discontinuous - Po odeslání příkazu k zážehu zůstává na stejné pozici.

D Fire All - Odešle příkaz k zážehu všech připojených zápalnic.

Po vybrání jedné z možností se výběr potvrdí potvrzovacím tlačítkem. Pokud je vybrána možnost D text se změní na tvar: „D000“, v případě výběru A, B nebo C se označí číslo, které je nyní možno měnit bočními tlačítky. Toto číslo představuje výstup na přijímačích. Po nastavení požadovaného módu se může zážeh spustit sepnutím vypínače pro odjištění zážehu na pravé straně vysílače a následným stiskem potvrzovacího tlačítka. V případě sepnutí vypínače pro odjištění zážehu vysílač vyšle příkaz pro odjištění přijímačů a potvrzovací tlačítko se rozsvítí červeně. Další funkcí v menu, při posunutí menu horním bočním tlačítkem, je časovač. Nastavení času se opět provádí stisknutím horního tlačítka a následně lze měnit označenou pozici pomocí bočních tlačítek (začíná se na hodinách). Po nastavení požadované hodnoty na dané pozici se toto nastavení potvrdí horním tlačítkem a kurzor se posune na další pozici (minuty a následně sekundy). Potvrzením poslední pozice vysílač odešle čas do přijímačů. Poté lze odpočet spustit sepnutím vypínače pro odjištění zážehu a následným stiskem potvrzovacího tlačítka. Předposlední funkce v menu je



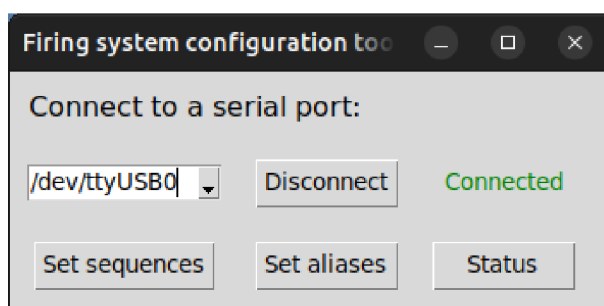
Obrázek 3.16: Blokové schéma funkce programu přijímače

nastavení „Settings“, zde je možné nastavit sekvence zážehu, Spreading Factor a vysílací výkon vysílače i přijímačů nebo přidělit výstupu na přijímači „přezdívku“. Poslední funkcí v menu jsou „Commands“, zde je možné vysílat testovací příkazy, dotazy na teplotu uvnitř přijímačů nebo stav baterie přijímačů. Návrat zpět do menu lze provést podržením spodního bočního tlačítka a následně stisknutím potvrzovacího tlačítka. Na displeji vysílače se v horním řádku nachází stavová lišta, která informuje o zbývajícím kapacitě baterie vysílače a síle přijímaného signálu.

Přijímač je naprogramován tak aby v zajištěném stavu ignoroval všechny příkazy k zážehu nebo spuštění odpočtu k zážehu a reagoval pouze na příkazy nastavení, testovací příkazy, dotazy na teplotu nebo stav baterie. Blokové schéma popisující funkci programu v přijímači je vyobrazeno na obrázku 3.16. V zajištěném stavu indikační LED na přijímači bliká zelenou barvou. Po přijetí příkazu na odjištění přijímače se rozezní bzučák a změní se barva blikání indikační LED na červenou. V tomto stavu je již možný zážeh. Po přijetí příkazu k zážehu se rozezní bzučák, indikační LED se rychle rozbliká bílou barvou a sepne se daný výstup. Po přijetí příkazu k zážehu všech výstupů, kdy není nastavená žádná sekvence, se sepnou všechny výstupy. V případě, kdy nastavená sekvence je, budou přijímače po přijetí příkazu k zážehu všech výstupů spínat výstupy dle nastavené sekvence. Vždy při každém zážehu se rozezní bzučák a rozbliká indikační LED bílou barvou. V případě přijetí příkazu pro zajištění přijímače se zruší všechny probíhající odpočty a sekvence. Pokud zbývajícím kapacita baterie přijímače klesne pod 30 %, je tento stav indikován bzučákem a rozsvícením LED indikátoru modrou barvou každých 30 sekund.

3.3.2 Počítačový program pro zadávání sekvencí

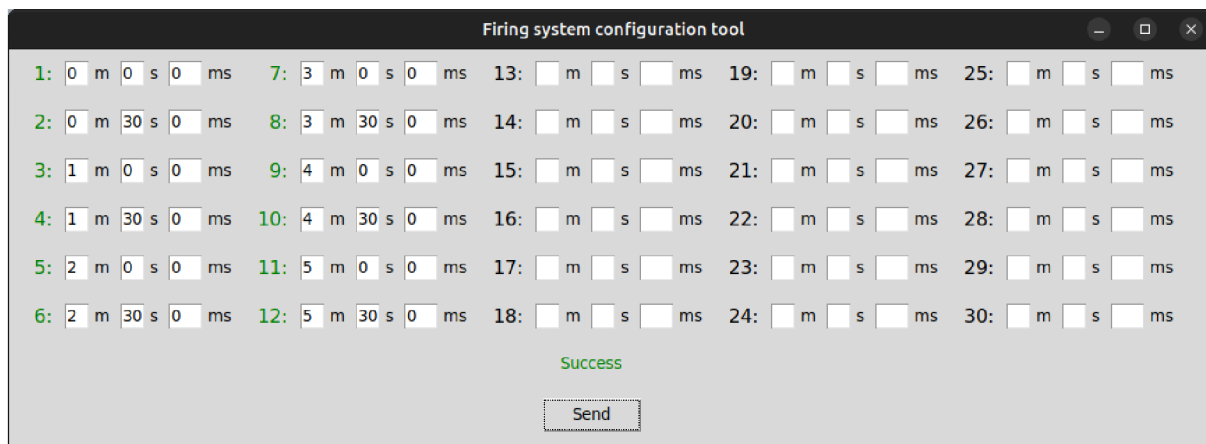
Tento počítačový program slouží pro pohodlné nastavení sekvencí zážehů a „přezdívek“ výstupů. Program taktéž obsahuje informační okno (obrázek 3.19), kde lze vidět teplota uvnitř přijímače, stav baterie přijímače, nastavené časy sekvence a zda je daný přijímač odjištěný nebo zajištěný. Tyto informace lze obnovit stisknutím tlačítka „Refresh“. V tomto okně lze také stisknutím tlačítka „LED & Buzzer Test“ otestovat indikační LED a bzučák.



Obrázek 3.17: Hlavní okno pro připojení k vysílači a otevření dalších oken programu

Pro připojení k vysílači a otevření ostatních oken programu slouží hlavní okno na obrázku 3.17, které se otevře po spuštění programu. Bez aktivního připojení k vysílači není možné otevřít ostatní okna a je tedy potřeba se nejprve připojit. Pro připojení je zapotřebí vysílač zapnout a připojit pomocí USB kabelu k počítači. Následně se v okně vybere požadovaný port a klikne na tlačítko „Connect“. Před kliknutím na toto tlačítko je zapotřebí stisknout a držet (po dobu připojování) potvrzovací tlačítko na vysílači, aby se ve vysílači aktivovala sériová linka. Po úspěšném připojení se tlačítko změní na „Disconnect“

a zobrazí se zelený text „Connected“, poté je možné otevřít ostatní okna.



Obrázek 3.18: Okno pro nastavení sekvencí zážehů

Stisknutím tlačítka „Set sequences“ se otevře okno pro nastavení sekvencí zážehů na obrázku 3.18. V tomto okně se nastavuje zpoždění sepnutí výstupu po přijetí příkazu pro zážeh všech výstupů u přijímače nebo po vypršení nastaveného odpočtu. Po přijetí tohoto příkazu nebo vypršení odpočtu bude tedy přijímač čekat nastavenou dobu u daného výstupu než dojde k jeho sepnutí. U políček kde není nastavena žádná hodnota dojde



Obrázek 3.19: Přehledové okno s informacemi o přijímačích

k sepnutí výstupu okamžitě. Nastavené hodnoty v tomto okně se odesílají do přijímačů pomocí tlačítka „Send“. Stisknutím tlačítka jsou hodnoty postupně odesílány do přijímačů a před odesláním další hodnoty vyčkává program na kontrolní odpověď. V případě, že se od přijímače vrátí chybná nebo žádná odpověď odesílání selže, vypíše se chybová hláška a číslo výstupu před hodnotou změní barvu na červenou. U hodnoty, která byla v pořádku odeslána a ověřena, se barva čísla změní na zelenou.

Po úspěšném nastavení sekvencí můžeme sekvenci spustit v přehledovém okně na obrázku 3.19. Přehledové okno se spouští pomocí tlačítka „Status“ na hlavním okně. Pro spuštění sekvence zážehu je nejprve zapotřebí odjistit přijímače pomocí sepnutí vypínače pro odjištění zážehu na vysílači. Po sepnutí tohoto vypínače změní tlačítko „Fire“ barvu z černé na červenou a přijímačům je odeslán příkaz na odjištění. Černě zabarvené tlačítko „Fire“ není možné stisknout. Přijímače po přijetí příkazu provedou odjištění a odešlou zpětnou odpověď. Program přijetím odpovědi reaguje změnou barvy tlačítka „LED & Buzzer Test“, které má funkci testovacího tlačítka pro indikátory a zároveň slouží jako indikátor stavu přijímače. Zelená barva tlačítka indikuje zajištěný stav a červená barva indikuje stav odjištěný. V případě úspěšného odjištění všechna testovací tlačítka přijímačů změní barvu na červenou a je možné zahájit zážeh pomocí tlačítka „Fire“.

Stisknutím tlačítka „Fire“ se vyšle příkaz k zážehu a na informačním okně započne odpočet sekvencí zážehů, jak je vidět na obrázku 3.20. Aktivní odpočty jsou zelenou barvou. Po vypršení se odpočet zbarví červeně.



Obrázek 3.20: Přehledové okno s informacemi o přijímačích při spuštěné odpočtu sekvencí

3.4 Ověření funkčnosti návrhu

3.4.1 Ověření funkčnosti PC programu

Počítačový program jsem otestoval na počítači s operačním systémem Ubuntu Mate ve verzi 22.04 s verzí Python 3.10 za použití dvou přijímačů u kterých byly k výstupům místo palníků připojeny LED. Při testování jsem nejprve zapnul vysílač otočením klíčku hlavního vypínače a připojil pomocí USB kabelu k počítači. Na počítači jsem spustil program a v hlavním okně vybral příslušný port „/dev/ttyUSB0“, jenž byl přiřazen připojenému vysílači. Před kliknutím na tlačítko „Connect“ v programu jsem na vysílači zmáčkl a po dobu připojování držel potvrzovací tlačítko. Po připojení programu k vysílači jsem zapnul přijímací jednotky. Komunikaci jsem ověřil otevřením přehledového okna pomocí tlačítka „Status“ na hlavním okně a stisknutím tlačítka „Refresh“. Po stisknutí tohoto tlačítka se u obou přijímačů zobrazili aktuální informace o jejich stavu. Zároveň jsem v tomto okně otestoval také indikační LED a bzučák pomocí tlačítek „LED & Buzzer Test“. Stisknutím tlačítka u prvního přijímače se rozezněl bzučák a rozsvítila indikační LED bílou barvou prvního přijímače. Stejně se tak stalo na druhém přijímači po stisknutí tlačítka druhého přijímače. Poté jsem toto okno zavřel a pomocí tlačítka „Set sequences“ na hlavním okně jsem otevřel okno pro nastavení sekvencí. Zde jsem zadal sekvence spínání po 30 sekundách a odeslal do přijímačů pomocí tlačítka „Send“. Nastavení bylo úspěšně přeneseno do přijímačů, jelikož se všechna čísla zbarvila zeleně a došlo k vypsání zprávy „Success“ jako na obrázku 3.18. Nyní jsem opět otevřel přehledové okno, odjistil přijímače pomocí vypínače pro odjištění zážehů na vysílači. Poté co testovací tlačítka změnila barvu ze zelené na červenou, která značí odjištění přijímače, jsem stiskl tlačítko „Fire“. Stisknutím tohoto tlačítka započalo odpočítávání u jednotlivých výstupů a okno nyní vypadalo jako na obrázku 3.20. Po vypršení odpočtu u výstupu v programu se u daného výstupu na přijímači aktivovala připojená LED.

3.4.2 Spolehlivost bezdrátové komunikace

Spolehlivost bezdrátové komunikace jsem ověřil na vzdálenost přibližně 425 metrů s přímým výhledem na přijímač, při maximálním vysílacím výkonu a SF nastaveným na hodnotu 7. Při testování na této vzdálenosti nedošlo ke ztrátě žádného z vyslaných paketů a síla přijímaného signálu od přijímače se pohybovala v rozmezí od -92 dBm do -98 dBm.

3.4.3 Posouzení z hlediska bezpečnosti

Na přijímače jsem se pokoušel, v zapnutém a zajištěném stavu, vysílat příkazy k zážehu nebo spuštění odpočtu. Přijímače se při testování zachovali podle očekávání, tyto příkazy ignorovali a výstupy nesejnuli. Taktéž jsem zkoušel nastavit odpočty a sekvence zážehu. Po odjištění a spuštění odpočtu nebo sekvence jsem zjišťoval zda se odpočet nebo sekvence po zajištění přeruší. Při testech se odpočty a sekvence, po zajištění přijímačů, přerušili a nedošlo k sepnutí výstupů.

3.4.4 Zkouška výdrže baterie

Přijímače s plně nabitou baterií vydrželi při testování za pokojové teploty 3,5 hodiny. Během této doby probíhala komunikace mezi přijímači a vysílačem zhruba každých 15 minut z důvodu sledování zbývající kapacity baterie a teploty uvnitř přijímače. Při testování jsem zjistil nadměrné zahřívání uvnitř přijímače, které podle vnitřního teploměru dosahovalo

teploty až 44°C. Po dalším prošetření jsem zjistil, že dané zahřívání způsobuje především regulátor zajišťující napájení indikační LED a bzučáku. Toto teplo naštěstí neproniká k baterii a nezpůsobuje její zahřívání. Nicméně výměnou tohoto regulátoru za účinnější by se mohla prodloužit doba provozu.

Výdrž vysílače, na plně nabitou baterii, nebyla přesně testována. Nicméně, podle rychlosti vybíjení, odhaduji výdrž s plně nabitou baterií na více než 10 hodin.

4 Závěr

Hlavním cílem této bakalářské práce byl návrh a realizace vlastního řešení dálkového systému pro zážeh zábavní pyrotechniky. Součástí práce byla také rešerše v oblasti dostupných řešení na českém trhu a technologií použitých k realizaci vlastního řešení.

Práce se zabývala návrhem schémat zapojení přijímače a vysílače, včetně tvorby DPS v programu KiCad. Následným návrhem krabiček, odolných proti vniknutí prachu a vlhkosti, pomocí CAD programu SolidWorks a jejich zhotovením pomocí technologie 3D tisku. V neposlední řadě bylo sestavení výsledného návrhu, jehož části jsou v práci podrobně popsány.

Vytvořený návrh dálkového systému byl naprogramován s využitím platformy Arduino, s ohledem na bezpečnost a pohodlnou obsluhu uživatelem. K tomuto systému byla v jazyce Python naprogramována počítačová aplikace pro zadávání sekvencí zážehů a dohled nad systémem. V této práci byly jednotlivé funkce systému, včetně popisu funkce počítačového programu, popsány.

Celý systém byl nakonec testován, aby byla ověřena jeho funkčnost a bezpečnost, podle kritérií daných v této práci. Testování objevilo nedostatky, který má vliv na výdrž baterie v přijímači a bylo navrženo řešení tohoto nedostatku. V ostatních ohledech testování prokázalo funkčnost a bezpečnost systému pro používání.

Literatura

- [1] REMOTE CONTROL EVERYTHING. *Firework Ignitor System for launching 2 Electric Igniters Remotely* [online]. Remote Control Everything, 2013-05-13. [cit. 2023-05-08]. Dostupné z: <http://www.remotecontroleverything.com/topics/special-remote-control/remote-control-fireworks/page/2/>.
- [2] CHILI ROSES.CZ. *Bezdrátové odpalovací zařízení pro 8 ODPALŮ* [online]. CHILI ROSES.CZ, s.r.o., b.r. [cit. 2023-05-13]. Dostupné z: <https://www.zabavni-pyrotechnika.cz/p/204/bezdratove-odpalovaci-zarizeni-pro-8-odpalu>.
- [3] ONDRÁK, Vladimír. *PALNÍKY* [online]. Ohňostroje Ondrák - kompletní realizace ohňostrojů a prodej zábavní pyrotechniky, ©2011–2014. [cit. 2023-05-08]. Dostupné z: <https://www.ohnostroje-ondrak.cz/Palniky.html>.
- [4] ZVONEK, Alan. *Palníky* [online]. OHŇOSTROJE-ZVONEK, ©2008–2020. [cit. 2023-05-14]. Dostupné z: <https://ohnostroje-zvonek.cz/index.php/palniky>.
- [5] PYROSTAV.CZ. *Odpalovací zařízení ALPHA FIRE 1 odpal* [online]. Pyrostav.cz, ©2023. [cit. 2023-05-13]. Dostupné z: <https://www.pyrostav.cz/odpalovaci-zarizeni-alpha-fire-1-odpal-p198>.
- [6] PYROSTAV.CZ. *Odpalovací zařízení ALPHA FIRE 4 odpaly* [online]. Pyrostav.cz, ©2023. [cit. 2023-05-13]. Dostupné z: <https://www.pyrostav.cz/odpalovaci-zarizeni-alpha-fire-4-odpaly-p199>.
- [7] PYROSTAV.CZ. *Odpalovací zařízení ALPHA FIRE 12 odpalů* [online]. Pyrostav.cz, ©2023. [cit. 2023-05-13]. Dostupné z: <https://www.pyrostav.cz/odpalovaci-zarizeni-alpha-fire-12-odpalu-p200>.
- [8] CHILI ROSES.CZ. *Bezdrátové odpalovací zařízení na 24 odpalů* [online]. CHILI ROSES.CZ, s.r.o., b.r. [cit. 2023-05-13]. Dostupné z: <https://www.zabavni-pyrotechnika.cz/p/401/odpalovaci-zarizeni-na-24-odpalu>.

- [9] IGNISSHOP.COM. *Odpalovací zařízení* [online]. Ignisshop.com, ©2020. [cit. 2023-05-13]. Dostupné z: <https://ignisshop.com/cs/pyrotechnicke-zarizeni/119-odpalovaci-zarizeni.html>.
- [10] SLANINA, Martin. *Moderní bezdrátová komunikace : přednášky*. Vyd. 1. V Brně: Vysoké učení technické v Brně, Fakulta elektrotechniky a informatiky, Ústav radioelektroniky, 2010. ISBN 978-80-214-4156-9.
- [11] MOLISCH, Andreas F. *Wireless communications*. 2nd ed. Chichester: John Wiley & Sons, 2011. ISBN 978-0-470-74186-3.
- [12] LIANDO, Jansen; GAMAGE, Amalinda; TENGOURTIUS, Agustinus; LI, Mo. Known and Unknown Facts of LoRa: Experiences from a Large-scale Measurement Study. *ACM transactions on sensor networks*. 2019, roč. 15, č. 2, s. 1–35. ISSN 1550-4859.
- [13] AI-THINKER TECHNOLOGY. *Ra-02 LoRa Module* [online]. ©2017. [cit. 2023-05-05]. Dostupné z: https://docs.ai-thinker.com/_media/lora/docs/c048ps01a1_ra-02_product_specification_v1.1.pdf.
- [14] DUDÁČEK, Karel. *Mikrokontroléry* [online]. 2001. [cit. 2023-04-30]. Dostupné z: <http://home.zcu.cz/~dudacek/Pot/mikrokontrolery.pdf>.
- [15] MICROCHIP. *ATmega328P* [online]. 2020. [cit. 2023-04-30]. Dostupné z: <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-DS40002061B.pdf>.
- [16] ARDUINO. *Arduino Pro Mini* [online]. Arduino, ©2023. [cit. 2023-05-14]. Dostupné z: <https://docs.arduino.cc/retired/boards/arduino-pro-mini>.
- [17] SPARKFUN ELECTRONICS. *FTDI Basic Breakout - 3.3V* [online]. SparkFun Electronics, b.r. [cit. 2023-05-16]. Dostupné z: <https://www.sparkfun.com/products/retired/8772>.
- [18] FTDI. *FT232R USB UART IC Datasheet* [online]. 2020-05-21. [cit. 2023-05-06]. Dostupné z: https://ftdichip.com/wp-content/uploads/2020/08/DS_FT232R.pdf.
- [19] VOREL, Pavel; PATOČKA, Miroslav. *Průmyslová elektronika*. Brno: Vysoké učení technické, 2007.

- [20] ADVANCED MONOLITHIC SYSTEMS. *AMS1117 1A LOW DROPOUT VOLTAGE REGULATOR* [online]. 2009-08-27. [cit. 2023-05-15]. Dostupné z: <http://www.advanced-monolithic.com/pdf/ds1117.pdf>.
- [21] LASKAKIT. *Nabíječka Li-ion článku TP4056 s ochranou microUSB* [online]. LaskaKit, ©2023. [cit. 2023-05-15]. Dostupné z: <https://www.laskakit.cz/nabijecka-li-ion-clanku-tp4056-s-ochranou-microusb/>.
- [22] LASKAKIT. *Nabíječka Li-ion článku TP4056 microUSB* [online]. LaskaKit, ©2023. [cit. 2023-05-15]. Dostupné z: <https://www.laskakit.cz/nabijecka-li-ion-clanku-tp4056-microusb/>.
- [23] HUNT, John. *A Beginners Guide to Python 3 Programming*. Cham: Springer International Publishing AG, 2019. Undergraduate topics in computer science. ISBN 3030202895.
- [24] LIECHTI, Chris. *Welcome to pySerial's documentation* [online]. pySerial, ©2001–2020. [cit. 2023-05-01]. Dostupné z: <https://pyserial.readthedocs.io/en/latest/index.html#>.
- [25] CHARATAN, Quentin; KANS, Aaron. Python Graphics with Tkinter. In: *Programming in Two Semesters*. Switzerland: Springer International Publishing AG, 2022. ISBN 3031013255.
- [26] PYTHON TUTORIAL.NET. *Tkinter Grid Geometry Manager* [online]. pythontutorial.net, ©2023. [cit. 2023-04-30]. Dostupné z: <https://www.pythontutorial.net/tkinter/tkinter-grid/>.
- [27] ROSEMAN, Mark. *TkDocs Tutorial - The Grid Geometry Manager* [online]. TkDocs, ©2007–2020. [cit. 2023-05-01]. Dostupné z: <https://tkdocs.com/tutorial/grid.html>.
- [28] GREATSCOTT. *DIY LiPo Charge/Protect/5V Boost Circuit*. In: *YouTube* [online]. 2017-09-10 [cit. 2023-05-06]. Dostupné z: <https://youtu.be/Fj0XuYiE7HU>.
- [29] NANJING TOP POWER ASIC CORP. *TP4056 1A Standalone Linear Li-Ion Battery Charger with Thermal Regulation in SOP-8* [online]. 2008-05-04. [cit. 2023-05-07]. Dostupné z: <https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf>.

- [30] INFINEON TECHNOLOGIES AG. *IRLR/U3103PbF* [online]. 2011-05-07. [cit. 2023-05-12]. Dostupné z: https://www.infineon.com/dgdl/Infineon-IRLR3103-DataSheet-v01_01-EN.pdf?fileId=5546d462533600a40153566ce4382684.
- [31] MISTRY, Sandeep. *Arduino LoRa* [online]. GitHub, Inc., ©2016. [cit. 2023-05-05]. Dostupné z: <https://github.com/sandeepmistry/arduino-LoRa>.
- [32] ADAFRUIT INDUSTRIES. *Adafruit SSD1306* [online]. GitHub, Inc., ©2012. [cit. 2023-05-19]. Dostupné z: https://github.com/adafruit/Adafruit_SSD1306.
- [33] ADAFRUIT INDUSTRIES. *Adafruit GFX Library* [online]. GitHub, Inc., ©2012. [cit. 2023-05-19]. Dostupné z: <https://github.com/adafruit/Adafruit-GFX-Library>.
- [34] ADAFRUIT INDUSTRIES. *Adafruit Bus IO Library* [online]. GitHub, Inc., ©2017. [cit. 2023-05-19]. Dostupné z: https://github.com/adafruit/Adafruit_BusIO.
- [35] ARDUINO. *Libraries* [online]. Arduino, ©2023. [cit. 2023-05-25]. Dostupné z: <https://www.arduino.cc/reference/en/libraries/>.

Seznam zkratek

BW	Bandwidth
CAD	Computer-Aided Design
CC	Constant Current
CR	Code Rate
CSS	Chirp Spread Spectrum
CV	Constant Voltage
DMA	Direct Memory Access
DPS	Deska plošných spojů
EEPROM	Electrically Erasable Programmable Read-Only Memory
GUI	Graphical User Interface
I/O	Input/Output
IoT	Internet of Things
ISM	Industrial, Scientific and Medical radio
LDO	Low Dropout
LED	Light-Emitting Diode
Li-Pol	Lithium-polymer
LoRa	Long Range
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
OS	Operating system
PC	Personal computer
PCB	Printed Circuit Board
PETG	Polyethylene terephthalate glycol
SF	Spreading Factor
SFD	Start Frame Delimiter

TPE Thermoplastic elastomer

UART Universal Asynchronous Receiver-Transmitter

USB Universal Serial Bus

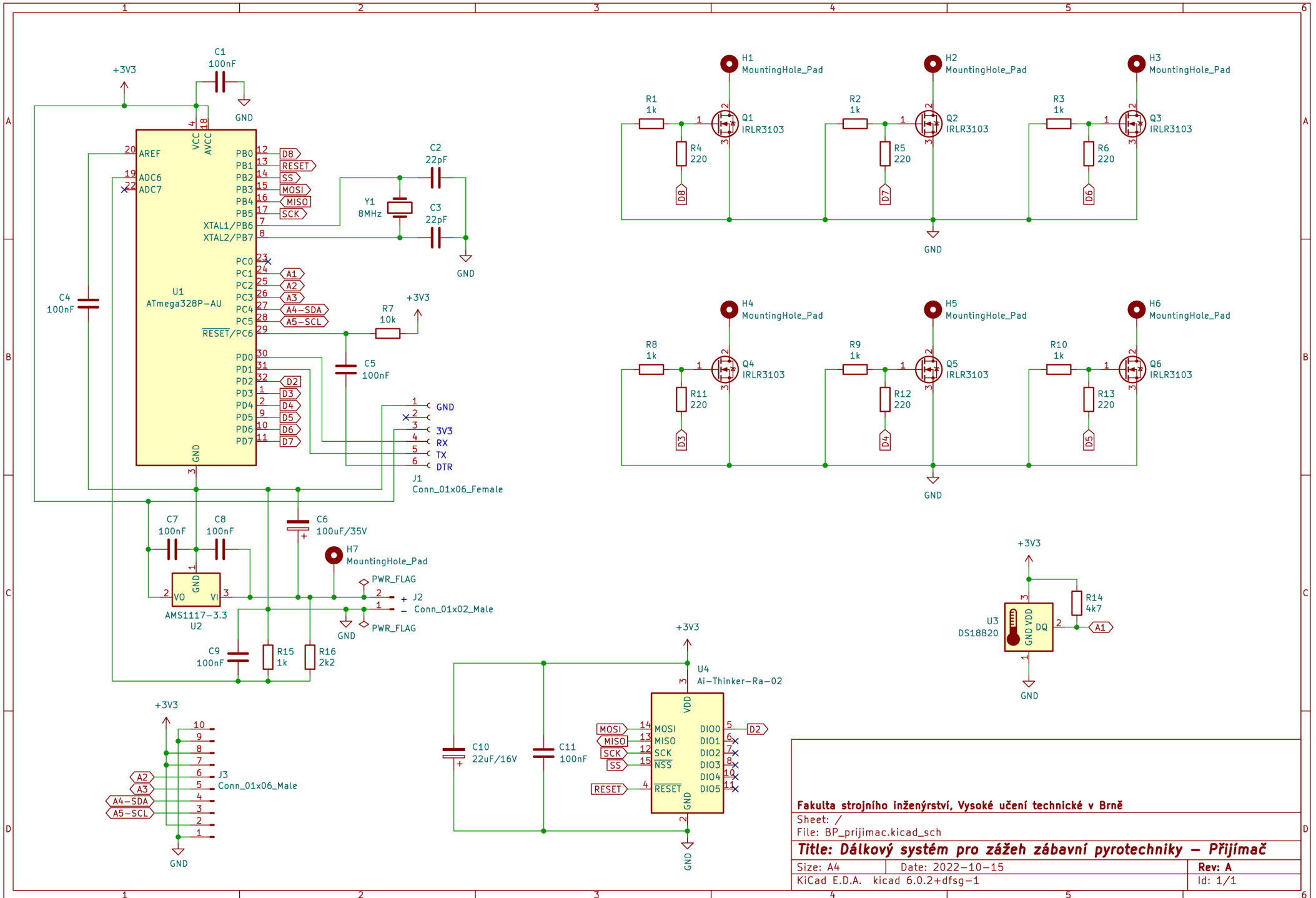
Seznam příloh

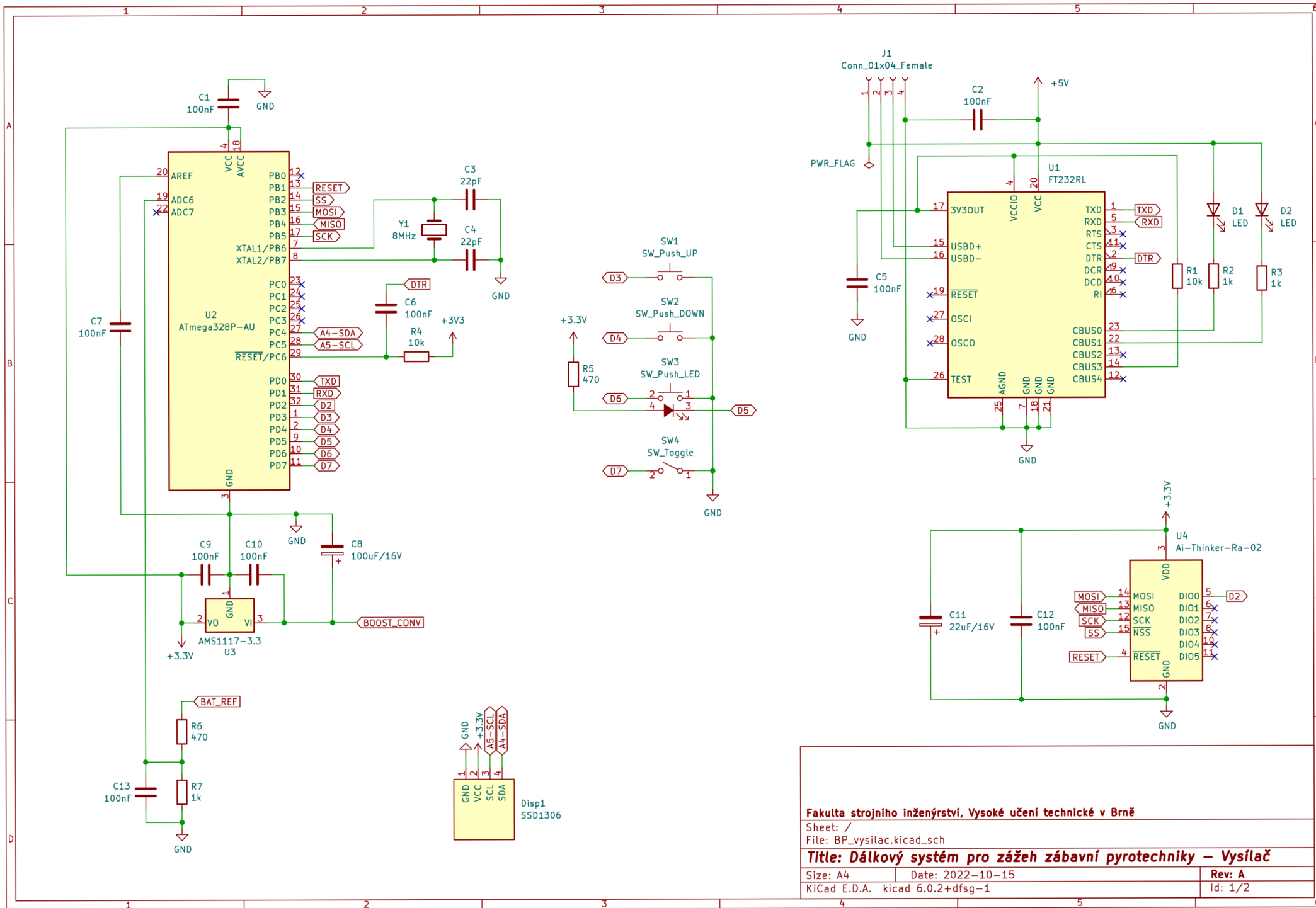
Přílohy uvedené v této práci

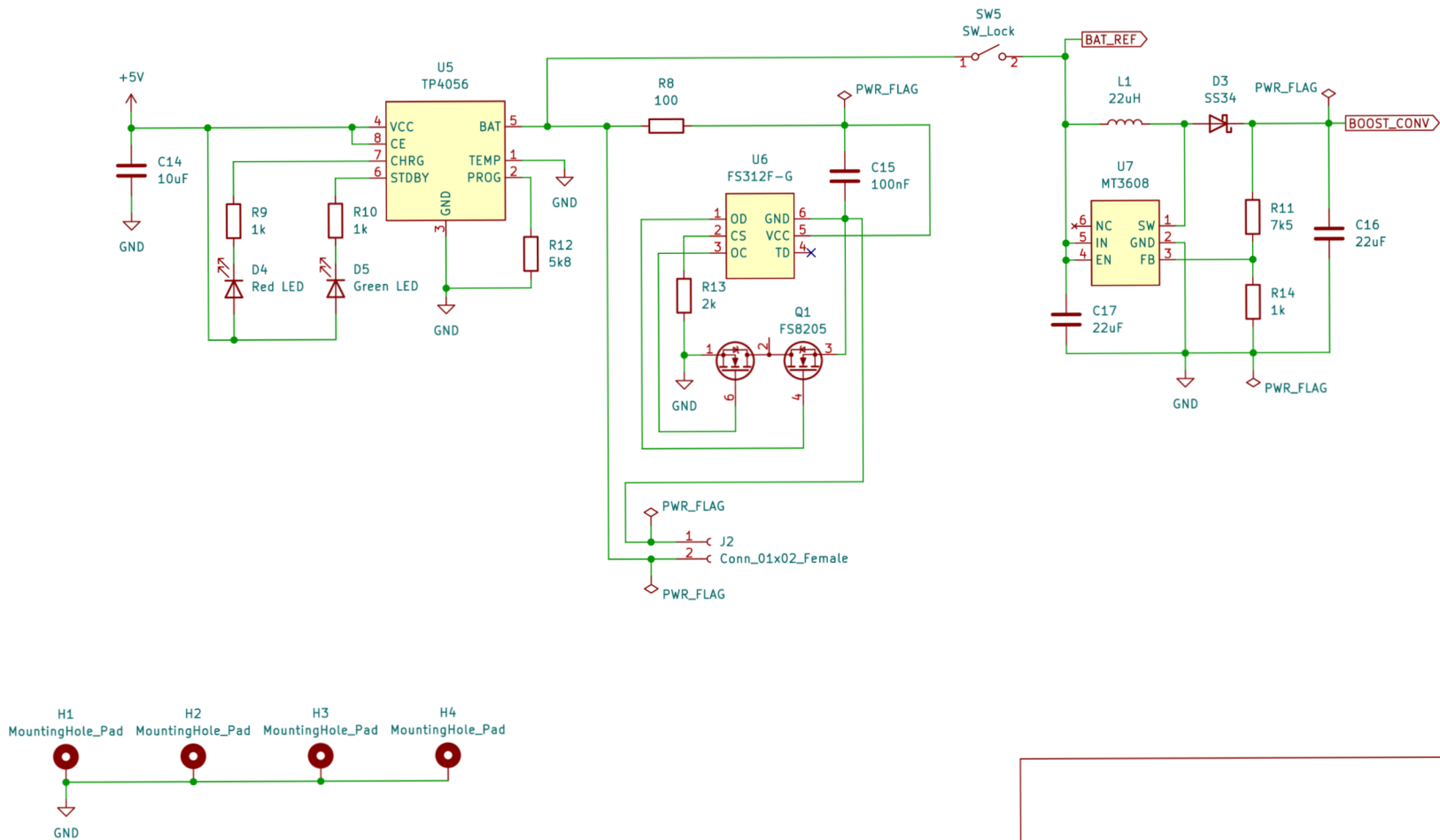
- Schéma zapojení přijímače
- Schéma zapojení vysílače

Přílohy uvedené v externím souboru

- „3D_modely“ - 3D modely krabiček ve formátu STEP a SLDPRT (SolidWorks 2020).
- „DPS“ - Soubory návrhů DPS programu KiCad, Gerber soubory pro výrobu DPS a schémata zapojení.
- „Firmware“ - Zdrojový kód Arduino pro vysílač a přijímač.
- „PC_Program“ - Skript Python 3 počítačového programu pro zadávání sekvencí.







Fakulta strojního inženýrství, Vysoké učení technické v Brně

Sheet: /Vysílač_BMS/

File: Vysilac_BMS.kicad_sch

Title: Dálkový systém pro zážeh zábavní pyrotechniky – Vysílač

Size: A4

Date: 2022-10-15

Rev: A

KiCad E.D.A. kicad 6.0.2+dfsg-1

Id: 2/2