

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A
INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2025

Ondřej Vozáb

Univerzita Pardubice
Fakulta Elektrotechniky a Informatiky

Modulační a řídicí jednotka světelné a zvukové techniky s rozhraním DMX512
Bakalářská práce

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2024/2025

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Ondřej Vozáb**
Osobní číslo: **I22036**
Studijní program: **B0714A150008 Automatizace**
Téma práce: **Modulační a řídicí jednotka světelné a zvukové techniky s rozhraním DMX512**
Zadávající katedra: **Katedra automatizace a matematiky**

Zásady pro vypracování

Cílem práce je vytvoření elektronické jednotky pro generování zvukových modulačních efektů (Reverb) s možností ovládání osvětlovací techniky s komunikačním protokolem DMX512. Návrh konstrukčního řešení jednotky a její realizace bude provedena na úrovni funkčního prototypu.

Teoretická část práce bude obsahovat podrobnou rešerši zadaného tématu. Z této rešerše bude vycházet návrh vlastního prototypu jednotky. Základem jednotky bude výpočetní jednotka s mikrokontrolérem STM32H750. Hlavní funkce mikropočítače bude zpracování zvuku a zajištění komunikace ve formátu komunikačního protokolu DMX512. Ovládání jednotky bude realizováno, například ovládacím panelem s vizualizačním displejem typu TFT, nebo nadřazeným řídicím softwarem osobního počítače. Jednotka bude umožňovat ovládání komerčně dostupné osvětlovací techniky s rozhraním DMX512.

V praktické části práce bude proveden podrobný popis realizace navržené konstrukce prototypu elektronické jednotky, tvorby firmware mikropočítače a provedeno její testování. Testování bude realizováno v kombinaci laboratorních a reálných provozních podmínek. Výsledky testování, společně s výrobními podklady jednotky, budou součástí kompletní elektronické dokumentace práce.

Rozsah pracovní zprávy: **50**
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

MAZIDI, Muhammad Ali; CHEN, Shujen a GHAEMI, Eshragh. *STM32 Arm programming for embedded systems: using C language with STM32 Nucleo*. [Velká Británie]: MicroDigital, 2014-2018. ISBN 978-0-997-92594-4.

ZÁHLAVA, V., *Návrh a konstrukce DPS, BEN-technická literatura*, 2010, ISBN 978-80-7300-266-4

Vedoucí bakalářské práce: **Ing. Libor Havlíček, Ph.D.**
Katedra automatizace a matematiky

Datum zadání bakalářské práce: **15. prosince 2024**
Termín odevzdání bakalářské práce: **16. května 2025**

prof. Ing. Petr Doležel, Ph.D. v.r.
děkan

L.S.

Ing. Libor Kupka, Ph.D. v.r.
vedoucí katedry

V Pardubicích dne 24. ledna 2025

Prohlašuji:

Práci s názvem Modulační a řídicí jednotka světelné a zvukové techniky s rozhraním DMX512 jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 7/2019 Pravidla pro odevzdávání, zveřejňování a formální úpravu závěrečných prací, ve znění pozdějších dodatků, bude práce zveřejněna prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 12. 5. 2025

Ondřej Vozáb

PODĚKOVÁNÍ

Děkuji Ing. Liboru Havlíčkovi, Ph.D. za pomoc při vedení bakalářské práce. Mé poděkování patří též mým rodičům za financování práce. Také bych chtěl poděkovat Robinovi Klose za zpřístupnění 3D tiskárny a poskytnuté informace týkající se 3D tisku.

ANOTACE

Zvukové efekty a synchronizace osvětlení s hudebním signálem jsou klíčové prvky moderních audiovizuálních instalací. Cílem této práce je vytvořit elektronickou jednotku, která bude provádět analýzu zvukového signálu, identifikovat dominantní frekvenci a její amplitudu a na jejich základě ovládat barvu a intenzitu světel pomocí protokolu DMX512. Dále jednotka implementuje modulaci analogového signálu ve formě Schroederova reverbu.

KLÍČOVÁ SLOVA

digitální zpracování signálu, reverb, DMX512, synchronizace světla a zvuku, vizualizace hudby

TITLE

Modulation and Control Unit for Lighting and Audio Technology with DMX512 Interface

ANNOTATION

Sound effects and synchronization of lighting with a musical signal are key elements of modern audiovisual installations. The aim of this work is to create an electronic unit that will analyze the audio signal, identify the dominant frequency and its amplitude, and use this information to control the color and intensity of lights via the DMX512 protocol. Furthermore, the unit implements analog signal modulation in the form of Schroeder reverb.

KEYWORDS

digital signal processing, reverb, DMX512, light and sound synchronization, music visualization

OBSAH

SEZNAM ILUSTRACÍ A TABULEK.....	10
SEZNAM ZKRATEK A ZNAČEK	11
TERMINOLOGIE	12
ÚVOD.....	13
1 TEORETICKÁ ČÁST	14
1.1 Základy zpracování audiosignálu	15
1.2 Analogově-digitální převodníky (ADC).....	15
1.2.1 Vzorkování.....	15
1.2.2 Kvantování.....	16
1.3 Fourierova transformace	17
1.3.1 Diskrétní Fourierova transformace (DFT).....	18
1.3.2 Rychlá Fourierova transformace (FFT)	19
1.4 Protokol I2S	20
1.5 Protokol DMX512	21
1.6 Digitální efekt reverb a Schroederův algoritmus.....	22
2 PRAKTICKÁ ČÁST	24
2.1 Návrh a koncepce systému	25
2.2 Popis použitých komponent.....	27
2.2.1 PMOD I2S2	27
2.2.2 Obvody CD74HC126	28
2.2.3 STM32F407G-DISC1.....	28
2.2.4 Arduino UNO R3.....	29
2.2.5 Převodník MAX485.....	30
2.2.6 Osvětlovací zařízení Eurolite LC4.....	31
2.2.3 Napájecí modul LM2596	31
2.2.6 Převodník FT232RL	32
2.3 Realizace hardware	33
2.3.1 Zapojení hlavních částí systému	33
2.3.2 Fyzické propojení a montáž.....	34
2.3.3 Napájení systému	36

2.3.4 Testování základní funkčnosti	36
2.4 Realizace software	37
2.4.1 Inicializace periférií pro ovládání světél.....	38
2.4.2 FFT analýza a výběr dominantní frekvence	39
2.4.3 Ovládání světél – Arduino přijímač.....	42
2.4.4 Aplikace pro monitorování a ovládání presetů	44
2.4.5 Tvorba reverbu.....	46
3 ZÁVĚR	48
ZDROJE ELEKTRONICKÉ	50
SEZNAM PŘÍLOH.....	51

SEZNAM ILUSTRACÍ A TABULEK

Obr. 1.: Vizualizace vzorkování a kvantizace analogového signálu	16
Obr. 2.: Vizuální interpretace FFT [zdroj: nti-audio.com]	19
Obr. 3.: Časový diagram I2S protokolu [zdroj: allaboutcircuits.com]	20
Obr. 4.: Časový diagram protokolu DMX512 [zdroj: researchgate.net]	22
Obr. 5.: Vizualizace reverbu v prostoru.....	23
Obr. 6.: Blokové schéma systému pro ovládání světel pomocí protokolu DMX512	26
Obr. 7.: Blokové schéma Schroederova reverbu	26
Obr. 8.: PMOD I2S2 [zdroj: rs-online.com]	27
Obr. 9.: STM32F407G-DISC1 [zdroj: dnatechindia.com]	29
Obr. 10.: ArduinoUno R3 [zdroj: arduino.cc]	30
Obr. 11.: Převodník MAX485 [zdroj: laskakit.cz]	30
Obr. 12.: Eurolite LC4 [zdroj: Thomann.de]	31
Obr. 13.: Napájecí step-down modul LM2596 [zdroj: laskakit.cz]	32
Obr. 14.: Převodník FT232RL [zdroj: botland.cz]	33
Obr. 15.: Schéma zapojení hlavní části systému	34
Obr. 16.: Schéma zapojení napájecí části systému	36
Obr. 17.: Vývojový diagram hlavního programu STM32 pro ovládání světel.....	38
Obr. 18.: Vývojový diagram funkce FFT	40
Obr. 19.: Ukázka výpisu funkce FFT	41
Obr. 20.: Vývojový diagram hlavního programu pro Arduino Uno	42
Obr. 21.: Vývojový diagram procesování přijatých dat pro Arduino	43
Obr. 22.: Vývojový diagram hlavního kódu aplikace.....	45
Obr. 23.: Vývojový diagram funkce port_DataReceived	45
Obr. 24.: Hlavní kód programu pro tvorbu reverbu.....	46
Obr. 25.: Vývojový diagram funkce Reverb.....	47
Obr. 26.: Vývojové diagramy přerušení spuštěného po naplnění I2S DMA bufferu	47
Obr. 27.: Vrní pohled na finální konstrukci systému.....	49
Obr. 28.: Finální konstrukce DMX LED baru	49

SEZNAM ZKRATEK A ZNAČEK

ADC – Analog to Digital Converter

DAC – Digital to Analog Converter

DMA – Direct Memory Access

DMX – Digital Multiplex

DFT – Discrete Fourier Transform

DSP – Digital Signal Processing

FFT – Fast Fourier Transform

FPU – Floating Point Unit

HAL – Hardware Abstraction Layer

I2S – Inter-IC Sound

LED – Light Emitting Diode

MCLK – Master Clock

RAM – Random Access Memory

RGB – Red Green Blue

UART – Universal Asynchronous Receiver-Transmitter

USART – Universal Synchronous/Asynchronous Receiver-Transmitter

USB – Universal Serial Bus

TERMINOLOGIE

Aliasing: Jev vznikající při vzorkování signálu, kdy je vzorkovací frekvence nižší než dvojnásobek nejvyšší frekvence signálu. Vede ke zkreslení vzorkovaného signálu.

All-pass filtr: Typ digitálního nebo analogového filtru, který propouští všechny frekvence se stejnou amplitudou, ale mění jejich fázové charakteristiky.

Comb filtr: Filtr, jehož frekvenční odezva připomíná zuby hřebene; vzniká kombinací zpožděného signálu se samotným vstupem.

Dominantní frekvence: Frekvence s největší amplitudou v daném signálu nebo spektru; často odpovídá vnímanému tónu nebo hlavnímu prvku zvuku.

Preset: Předem uložené nastavení parametrů zařízení nebo softwaru, které lze rychle aktivovat bez nutnosti ruční konfigurace.

Reverb: Zvukový jev vznikající odrazy zvuku od povrchů v prostoru, které se slučují s původním zvukem a vytvářejí dojem dozvuku nebo prostoru.

ÚVOD

V dnešní době hrají audiovizuální technologie klíčovou roli v mnoha oblastech, od zábavního průmyslu až po umělecké instalace, koncertní produkce či interaktivní výstavy. Synchronizace světla a zvuku je zásadní pro vytvoření působivého dojmu a posílení atmosféry daného prostředí. Moderní osvětlovací a zvukové systémy umožňují nejen statické osvětlení, ale i dynamické změny barev, intenzity a efektů na základě hudebního signálu.

Jedním z nejrozšířenějších standardů pro řízení osvětlovací techniky je DMX512, což je protokol umožňující komunikaci mezi ovládacími jednotkami a světelnými zařízeními. Díky němu lze efektivně řídit jednotlivé světelné prvky, ať už jde o LED panely, reflektory nebo laserové efekty. Automatizace těchto systémů na základě analýzy zvukového signálu umožňuje synchronizaci vizuálního vjemu s hudbou, což se často využívá při koncertech, divadelních představeních nebo interaktivních instalacích.

Tato práce se zaměřuje na návrh a realizaci modulační a řídicí jednotky, která umožní analýzu zvukového signálu v reálném čase, identifikaci jeho dominantní frekvence, její amplitudy a následné ovládání světelných efektů prostřednictvím protokolu DMX512. Kromě řízení světelné techniky bude jednotka obsahovat také implementaci Schroederova reverbu, což je efekt používaný k simulaci dozvuku v akustických prostorech. Tento efekt přispěje k rozšíření možností aplikace tohoto zařízení.

Cílem této práce je tedy propojení zvukového a světelného zpracování do jednoho kompaktního systému, který najde uplatnění v různých audiovizuálních aplikacích. Bude zde popsán princip analýzy zvuku, návrh hardwarové i softwarové části jednotky a způsob jejího praktického využití. Výsledkem bude funkční prototyp, který umožní synchronizaci světelných efektů se zvukem a jeho modulací.

1 TEORETICKÁ ČÁST

Teoretická část bakalářské práce se zabývá problematikou synchronizace zvuku a osvětlení pomocí digitálního zpracování zvukového signálu a jeho převodu na řídicí parametry pro světelné efekty prostřednictvím protokolu DMX512. Důležitým aspektem je zpracování signálu v reálném čase, jeho frekvenční analýza a implementace zvukového efektu reverb.

Jedním ze stěžejních prvků tohoto systému je využití rychlé Fourierovy transformace (FFT) pro analýzu frekvenčního obsahu zvukového signálu. Fourierova transformace umožňuje převést signál z časové oblasti do frekvenční oblasti, což je nezbytné pro identifikaci dominantních frekvencí a jejich amplitud. Tyto hodnoty se následně používají pro generování parametrů řídicích světelných efektů. Zvukový signál je do systému přiváděn pomocí protokolu I2S, kde je následně analyzován a využit pro ovládání světelných zařízení.

Další důležitou součástí zpracování signálu je jeho vyhodnocení a přiřazení odpovídajících barevných hodnot RGB v závislosti na frekvenčních složkách. Tento proces zahrnuje definování frekvenčních pásem a jejich mapování na konkrétní barvy a intenzitu světla. Výstupní data jsou následně odesílána přes UART do převodníku DMX512, který zajišťuje jejich distribuci k jednotlivým světelným zařízením.

Kromě synchronizace zvuku a osvětlení tato práce zasahuje i do problematiky zvukového efektu reverb. Reverb je akustický jev, který se projevuje jako prodloužení zvuku v důsledku odrazů od povrchů v prostoru. V rámci digitálního zpracování signálu se reverb bude realizovat pomocí algoritmu Schroederova dozvuku na Obr. 7, který využívá kombinace zpožděných signálů a filtrů k simulaci tohoto efektu. Tato implementace byla zvolena z důvodu její efektivity a nenáročnosti na výpočetní zdroje.

Nedílnou součástí projektu je také uživatelské rozhraní v podobě počítačové aplikace. Aplikace slouží k nastavení parametrů pro řízení DMX512, kde uživatel může definovat barevnou reakci na jednotlivé frekvence a citlivost osvětlení na intenzitu vstupního signálu. Toto rozhraní usnadňuje konfiguraci systému a přizpůsobení osvětlení dle potřeb konkrétní aplikace.

1.1 Základy zpracování audiosignálu

Zvuk je mechanické vlnění, šířící se v látkovém prostředí, které je schopno vyvolat sluchový vjem. V případě průměrného člověka se jedná o vlnění v rozsahu od 16 Hz do 20 kHz. Pro tuto práci je jako hlavní zdroj signálu zvolena elektrická kytara, která pokud má 22 prahců a je ve standardním ladění (EADGHE), tak se její rozsah pohybuje od 82,41 Hz do 1174,66 Hz. Tento rozsah je ovšem pravdivý pouze v případě, pokud bychom zanedbali vyšší harmonické složky, které mohou sahát až do výše 6 kHz a pokud je kytarový signál ještě před vstupem do jednotky modulován zkreslujícími efekty, tyto složky mohou dosahovat výše 10 kHz. S ohledem k této úloze víme, že vyšší harmonické složky se při analýze nikdy nebudou projevovat, jako dominantní frekvence, ale přesto je musíme zohlednit při výběru komponent, jelikož při modulaci signálu za pomoci Schroederova reverbu chceme modulovat celé spektrum vstupního signálu.

1.2 Analogově-digitální převodníky (ADC)

Analogově-digitální převodník je zařízení, které převádí spojitý (analogový) signál na diskrétní (digitální) data. Tento proces je nezbytný pro jakékoli digitální zpracování signálu, jelikož většina reálných fyzikálních veličin, včetně zvuku, je analogová povahy. ADC pracuje na principu vzorkování a kvantování (vizualizace na Obr. 1).

1.2.1 Vzorkování

Vzorkování je proces, při kterém se analogový signál měří v pravidelných časových intervalech. Výsledkem je posloupnost hodnot, které reprezentují amplitudu signálu v konkrétních časových okamžicích. Frekvence, s jakou se tyto vzorky pořizují, se nazývá vzorkovací frekvence (f_s).

Podle Shannonova–Nyquistova teorému musí být vzorkovací frekvence alespoň dvojnásobkem nejvyšší frekvence obsažené v signálu, aby bylo možné signál jednoznačně rekonstruovat bez ztráty informace:

$$f_s \geq 2 * f_{max}$$

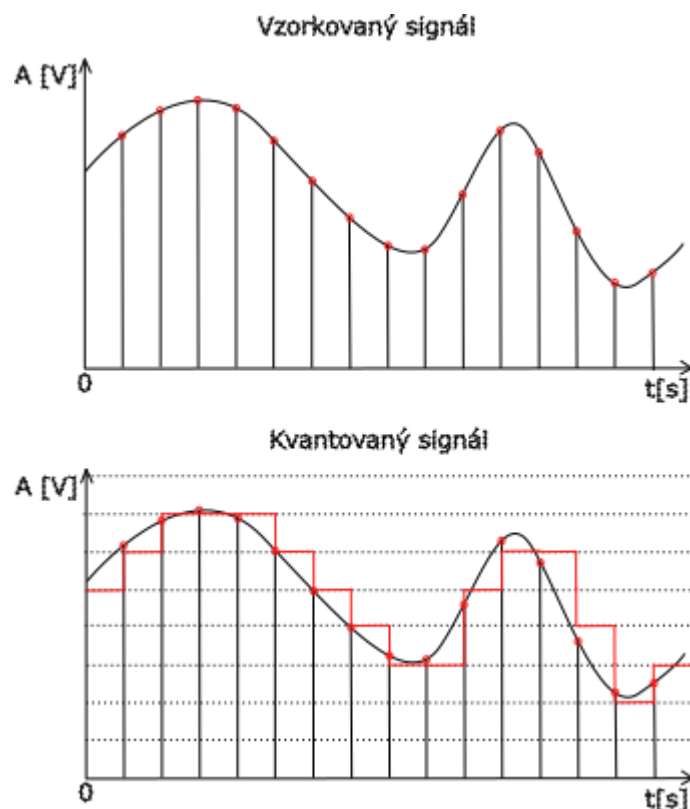
Pokud je vzorkovací frekvence příliš nízká, dochází k jevu zvanému aliasing, při kterém se vyšší frekvence překrývají a výsledný signál je zkreslený.

1.2.2 Kvantování

Kvantování je proces, při kterém se spojitá amplituda analogového vzorku převádí na diskrétní hodnotu, ta může být vyjádřena konečným počtem bitů. ADC má omezené rozlišení, které je dáno počtem bitů n , a tím pádem může rozlišit 2^n různých úrovní. Například 12bitový převodník dokáže rozlišit 2^{12} , tedy 4096 kvantizačních úrovní.

Rozsah napětí, které ADC dokáže měřit, se rozdělí rovnoměrně mezi tyto úrovně. Každý analogový vzorek se poté zaokrouhlí na nejbližší kvantizační úroveň. Tento krok zavádí určitou chybu, známou jako kvantizační šum. Čím více bitů ADC má, tím menší je kvantizační krok, a tedy i šum.

Kvantizační šum je definován jako rozdíl mezi skutečnou analogovou hodnotou a kvantizovanou hodnotou. Ve srovnání s hodnotami přenášeného signálu je ovšem natolik malý, že je většinou pro lidské ucho nezaznamatelný.



Obr. 1.: Vizualizace vzorkování a kvantizace analogového signálu

1.3 Fourierova transformace

Fourierova transformace představuje jeden ze základních nástrojů matematické analýzy používaný pro převod signálu z časové domény do domény frekvenční. Její vizuální interpretace se nachází na Obr. 2. Svůj název nese po francouzském matematikovi Josephu Fourierovi, který na počátku 19. století formuloval hypotézu, že každá periodická funkce může být vyjádřena, jako součet sinusových a kosinusových funkcí různých frekvencí, amplitud a fází. V dnešní době se stala naprosto nepostradatelnou, jelikož je uplatňovaná v celé řadě oborů, kterými jsou například:

- Zpracování signálů a zvuku – filtrování, spektrální analýza, komprese
- Zpracování obrazu – detekce hran, komprese, rozpoznávání
- Telekomunikace – modulace / demodulace

V kontextu této práce je Fourierova transformace při analýze audio signálu z elektrické kytary, kde slouží k detekci dominantní frekvence (tónu) a její amplitudy.

Fourierova transformace v základní podobě se používá pro spojité signály, které jsou definovány v nekonečném časovém intervalu. Jejím hlavním účelem je převod signálu z časové do frekvenční domény, čímž se získá informace o tom, jaké frekvenční složky signál obsahuje a v jaké míře je obsahuje.

Fourierova transformace spojité funkce $x(t)$ je definována jako:

$$X(f) = \int_{-\infty}^{\infty} x(t) * e^{-j2\pi ft} dt$$

Kde:

- $X(f)$ - spektrální reprezentace signálu
- f – frekvencí v Hz
- j – imaginární jednotka
- $e^{-j2\pi ft}$ – komplexní harmonické funkce

Výsledkem Fourierovi transformace je spektrum, kde každá frekvenční složka je reprezentována číslem, nesoucím informaci o její amplitudě a fázi.

1.3.1 Diskrétní Fourierova transformace (DFT)

V oblastech digitálního zpracování signálů (DSP), pracujeme s diskrétními signály. Tím myslíme vzorkované signály s konečným počtem hodnot. Pro tyto účely slouží diskrétní Fourierova transformace, která umožňuje analyzovat frekvenční složení signálu reprezentovaného konečným počtem vzorků.

Mějme posloupnost N vzorků signálu $x(n)$, kde $n = 0, 1, \dots, N - 1$. Potom je DFT definována vztahem:

$$X[k] = \sum_{n=0}^{N-1} x[n] * e^{-j2\pi \frac{kn}{N}}, \quad k = 0, 1, \dots, N - 1$$

Výsledkem je komplexní spektrum $X[k]$, kde každý index k odpovídá určité frekvenci:

$$f_k = \frac{k}{N} * f_s$$

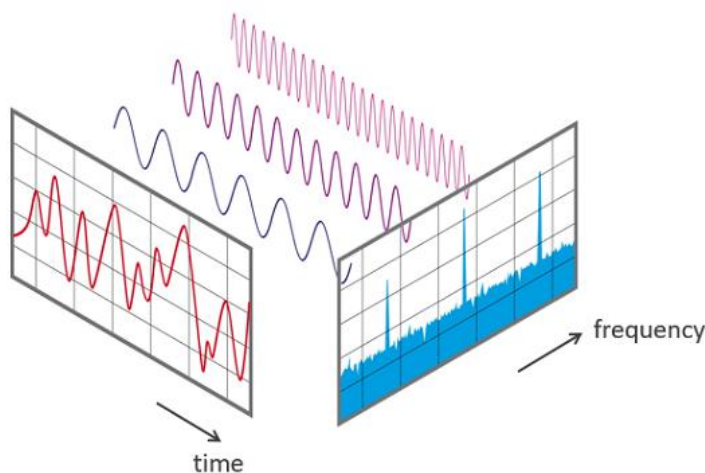
Kde f_s je vzorkovací frekvence.

DFT umožňuje analyzovat i neperiodické signály, a proto je základem mnoha digitálních technik včetně spektrografie, digitální filtrace, nebo identifikaci dominantních frekvencí v reálném čase.

1.3.2 Rychlá Fourierova transformace (FFT)

I když je DFT výpočetně velmi přesná, její praktické využití je omezeno zejména vysokou výpočetní náročností. Přímé výpočty DFT podle její definice vyžadují $\Theta(n^2)$ komplexních operací. Tato kvadratická závislost se stává velmi neefektivní v případech, kdy je potřeba zpracovat signály s o velkým počtu vzorků v reálném čase, nebo opakovaně v krátkých časových intervalech, což je typické pro aplikace zabývající se zpracováním zvuku, či obrazu.

Z důvodu této výpočetní náročnosti byla vyvinuta rychlá Fourierova transformace, která dramaticky zlepšuje výpočetní efektivitu. Její největší výhodou je výrazné snížení potřebných operací pro výpočet. Díky optimalizacím FFT potřebuje pro výpočet pouze $\Theta(n \log n)$ operací, což znamená, že počet výpočtů roste mnohem pomaleji než u klasické DFT. Tento rozdíl je z hlediska implementace ve zpracování signálu zcela zásadní, jelikož umožňuje frekvenční analýzu v reálném čase i výpočetně omezeným systémům.



Obr. 2.: Vizuální interpretace FFT [zdroj: nti-audio.com]

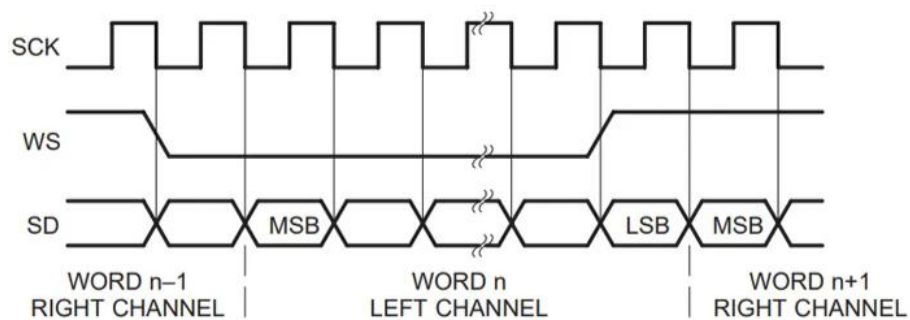
1.4 Protokol I2S

I2S (Inter-IC Sound) je sériový komunikační protokol, který je určen pro přenos digitálního audio signálu mezi integrovanými obvody. Standard I2S definuje rozhraní mezi zařízeními, jako jsou digitální audio převodníky (DAC/ADC), digitální signálové procesory (DSP), mikroprocesory nebo mikrořadiče, a je optimalizován pro přenos stereo audio dat ve formátu PCM (Pulse-Code Modulation).

Protokol I2S je navržen tak, aby minimalizoval jitter (kolísání časování) a zajistil přesnou synchronizaci při přenosu dat. Typická I2S komunikace využívá tři základní signálové linky:

- SD (Serial Data) – sériový přenos dat (většinou 16, 24 nebo 32 bitů)
- SCK (Serial Clock) – taktovací signál určující rychlost přenosu jednotlivých bitů
- WS (Word Select, někdy LRCK – Left/Right Clock) – přepíná mezi levým a pravým audiokanálem

Časový diagram I2S protokolu se nachází na Obr. 3:



Obr. 3.: Časový diagram I2S protokolu [zdroj: allaboutcircuits.com]

Kromě těchto tří linek může být použit i signál MCLK (Master Clock), který slouží k přesné synchronizaci systému. I2S je simplexní protokol, což znamená, že jeden kanál slouží buď pro přenos nebo příjem dat. Pro obousměrnou komunikaci je zapotřebí použít dvě nezávislé I2S linky.

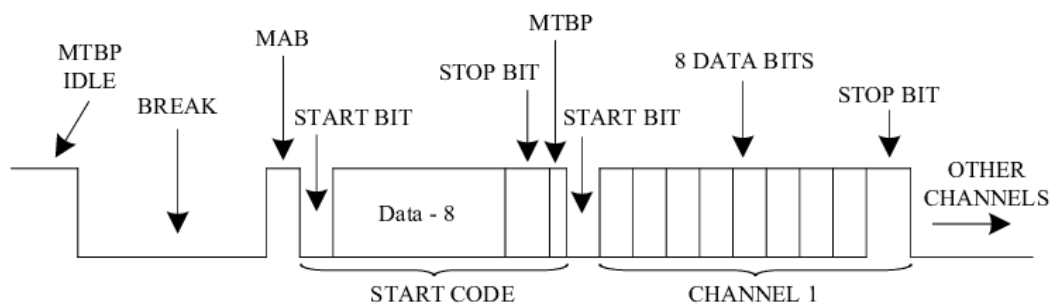
1.5 Protokol DMX512

Protokol DMX512 (Digital Multiplexer) je standardizovaný komunikační protokol určený pro řízení osvětlení jevištního osvětlení a efektových zařízení. Byl vyvinut organizací USITT (United States Institute for Theatre Technology) v roce 1986 a od té doby se stal standardem v oblasti divadelní, koncertní a televizní osvětlovací techniky.

DMX512 využívá sériovou digitální komunikaci na bázi standardu RS-485. Přenáší data po diferenciální dvou vodičové sběrnici, která zajišťuje vyšší odolnost proti okolnímu elektromagnetickému rušení a umožňuje přenos mezi zařízeními dlouhý až stovky metrů. Jedna DMX linka může přenášet až 512 kanálů, přičemž každý kanál reprezentuje jednu osmibitovou hodnotu v rozsahu 0–255. Tato hodnota obvykle odpovídá intenzitě světla, barvě nebo jinému parametru řízeného zařízení.

Komunikační rámec DMX512 je tvořen sekvencí bajtů, které jsou vysílány v pravidelných intervalech. Každý rámec začíná signálem zvaným break, což je úsek linky v logické nule po dobu alespoň 88 mikrosekund. Tento signál slouží jako indikace začátku nového rámce. Po něm následuje krátká pauza zvaná Mark After Break a poté startovací kód, který bývá u běžného DMX přenosu nulový (0x00). Za ním následují jednotlivé bajty odpovídající hodnotám jednotlivých kanálů – jejich celkový počet může být až 512. Tento rámec je neustále periodicky opakován, typicky s frekvencí okolo 44 Hz, čímž je zajištěna plynulá aktualizace hodnot. Vizualizace časového digramu protokolu DMX512 se nachází na Obr. 4.

Zařízení připojená na DMX sběrnici jsou adresována pomocí startovní adresy. Ta určuje, na který kanál dané zařízení začne reagovat. Například RGB LED světlo se startovací adresou 10 bude reagovat na kanály 10, 11 a 12, které obvykle ovládají červenou, zelenou a modrou složku světla. U některých zařízení lze také ovládat posun či rotace. Tímto způsobem lze na jediné sběrnici řídit mnoho různých zařízení s různým počtem kanálů. Velkou výhodou protokolu DMX512 je jeho jednoduchost a kompatibilita napříč zařízeními různých výrobců. Díky použití RS-485 je možné dosahovat přenosových vzdáleností i několika stovek metrů. Nevýhodou však je, že DMX512 je jednosměrný protokol – data tečou pouze od vysílače ke všem připojeným zařízením a neexistuje mechanismus zpětné vazby či potvrzení přijetí. Navíc každý kanál má pouze osmibitové rozlišení, což může být pro některé aplikace nedostačující.



Obr. 4.: Časový diagram protokolu DMX512 [zdroj: researchgate.net]

1.6 Digitální efekt reverb a Schroederův algoritmus

Reverb (dozvuk) je zvukový jev, který vzniká, když se akustický signál odráží od povrchů v prostoru, jako jsou stěny, stropy nebo podlahy (vizualizace reverbu v prostoru se nachází na Obr. 5), a výsledný signál se postupně mísí s přímým zvukem. Tento složený efekt vytváří dojem prostoru a hloubky. V akustických podmínkách se reverb přirozeně vyskytuje v každé místnosti, avšak v digitálním zpracování signálu se jedná o efekt, který je uměle generován za účelem simulace těchto prostorových vlastností, a to buď pro kreativní účely (např. ve zvukové produkci), nebo za účelem vylepšení vnímání nahraného signálu.

Digitální implementace reverbu je ve své podstatě aproximací tohoto přirozeného akustického jevu. Může být realizována různými způsoby. Jedním z algoritmičtých je například právě algoritmus navržený Manfredem Schroederem v 60. letech 20. století, který se stal jedním ze základních pilířů digitálního zpracování reverbu.

Schroederův algoritmus je založen na použití jednoduchých digitálních filtrů – konkrétně se jedná o kombinaci několika all-pass a comb filtrů, které dohromady vytvářejí efekt připomínající přirozený dozvuk v místnosti. Cílem tohoto algoritmu je vytvořit hustou a rovnoměrně rozloženou sérii ozvěn, které se postupně ztrácejí a tvoří tak přirozeně znějící dozvuk.

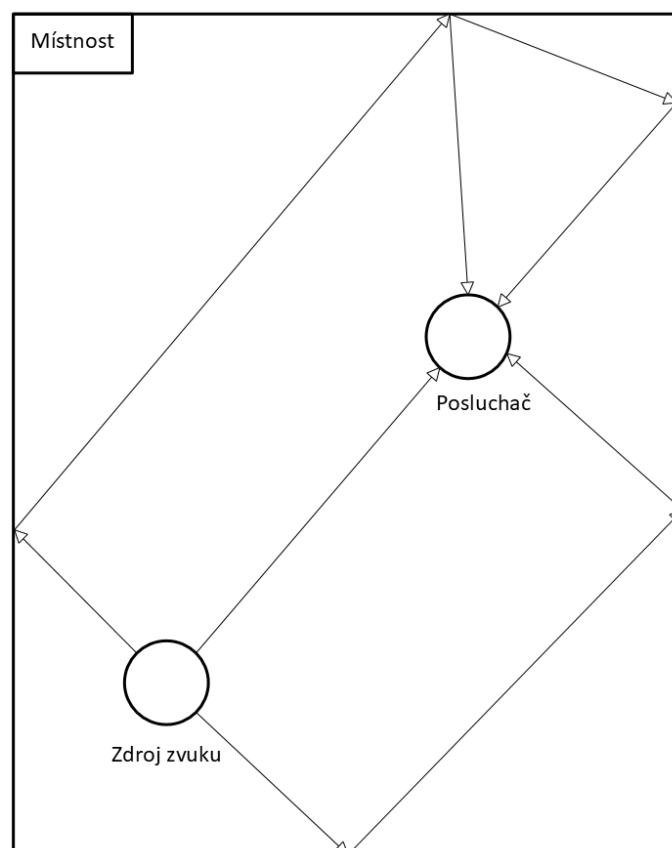
V první fázi algoritmu jsou použity paralelně zapojené feedback comb filtry (zpětnovazební hřebínkové filtry). Tyto filtry vytvářejí první reflexe – tedy rychlé a poměrně oddělené ozvěny, které se v reálném světě vyskytují v prvních desítkách milisekund po původním zvuku. Každý z těchto filtrů má jinou délku zpoždění, což zajišťuje rozmanitost v časech odrazů a zamezuje vzniku periodických artefaktů. Výsledný signál ze všech těchto filtrů se poté zprůměruje nebo jinak sloučí.

Následně se výstup z těchto kombinačních filtrů posílá do série několika all-pass filtrů. Tyto filtry samy o sobě nemění amplitudu signálu (mají jednotkový zisk), ale modifikují jeho

fázovou strukturu. V kontextu reverbu jsou využity ke zvyšování hustoty ozvěn a k rozptýlení energie dozvuku v čase, čímž se přispívá k přirozenějšímu výslednému zvuku. All-pass filtry se často zapojují sériově, což zvyšuje celkovou složitost výsledného reverbu, aniž by se přidávaly další odrazy s pevnou amplitudou.

Jedním z hlavních parametrů algoritmického reverbu je tzv. reverb time, což je doba, za kterou klesne úroveň doznívajícího signálu o 60 dB oproti původnímu impulsu. Tento parametr je možné řídit volbou zpětné vazby jednotlivých filtrů a délkou jejich zpoždění. Správné nastavení těchto hodnot je klíčové pro dosažení požadovaného vjemu prostoru – krátké časy simulují menší místnosti, zatímco delší časy odpovídají například koncertním sálům nebo katedrálám.

Výhodou Schroederova algoritmu je jeho relativní jednoduchost, nízká výpočetní náročnost a možnost snadné implementace v reálném čase, což jej činí ideálním pro použití při zpracování zvuku v reálném čase na DSP a mikrořadičích. Ačkoli dnes existují pokročilejší metody generování reverbu, algoritmus podle Schroedera zůstává dodnes respektovanou a používanou metodou, především jako základní stavební kámen nebo inspirace pro další algoritmické modely.



Obr. 5.: Vizualizace reverbu v prostoru

2 PRAKTICKÁ ČÁST

Úkolem praktické části této práce je návrh, realizace a ověření funkčnosti modulační a řídicí jednotky pro světelnou a zvukovou techniku s využitím protokolu DMX512. Na základě teoretických poznatků, popsaných v předchozí části, je nejprve vypracována koncepce celého systému včetně specifikace jednotlivých komponent a jejich vzájemného propojení.

V rámci realizace je detailně popsán výběr hardwarových a softwarových prostředků, návrh zapojení, postup implementace a konfigurace jednotlivých funkčních bloků, jako jsou analogově-digitální převod signálu, frekvenční analýza, řízení světelných pomocí DMX512 a implementace efektu reverb.

Zvláštní pozornost je věnována praktickému ověření funkčnosti vyvinutého zařízení. Probíhá testování v reálných podmínkách, měření a vyhodnocení dosažených výsledků. Výsledky jsou dále analyzovány a diskutovány, přičemž jsou rovněž navržena možná vylepšení a rozšíření systému.

2.1 Návrh a koncepce systému

Praktická část této práce je zaměřena na návrh a realizaci modulační a řídicí jednotky, která umožňuje analýzu zvukového signálu v reálném čase, řízení světelných efektů pomocí standardu DMX512 a digitální modulaci zvukového signálu za účelem vytvoření efektu reverb (dozvuku). Celé řešení kombinuje digitální zpracování signálu s řízením světelné techniky.

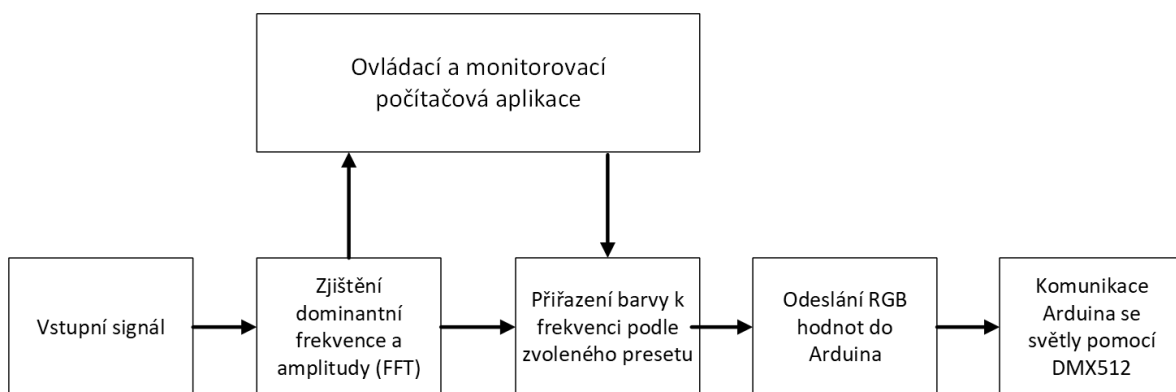
Pro získávání a digitalizaci zvukového signálu je využito modulu PMOD I2S2, který poskytuje vyšší rozlišení a kvalitu signálu oproti vestavěnému ADC převodníku, nacházejícího se v STM32F407G. Digitalizovaný signál je následně přenášen pomocí rozhraní I2S do mikrokontroléru STM32F407G, kde probíhá jeho zpracování. Hlavní zpracování signálu zahrnuje provedení rychlé Fourierovy transformace (FFT), která slouží k analýze frekvenčního obsahu signálu a k identifikaci dominantní frekvence a její amplitudy.

Po provedení spektrální analýzy jsou výsledky zpracovány v rámci interní logiky mikrokontroléru a převedeny na barevné informace typu RGB. Tyto barevné hodnoty odpovídají aktuálnímu frekvenčnímu obsahu vstupního signálu. Pro vizualizaci průběhu spektrální analýzy a nastavení provozních režimů systému je využíváno sériové rozhraní USART2, kterým STM32 komunikuje s externí počítačovou aplikací. Uživatel má prostřednictvím této komunikace možnost volit mezi čtyřmi přednastavenými režimy řízení světel (presety) a zároveň sledovat v reálném čase průběh frekvenční analýzy.

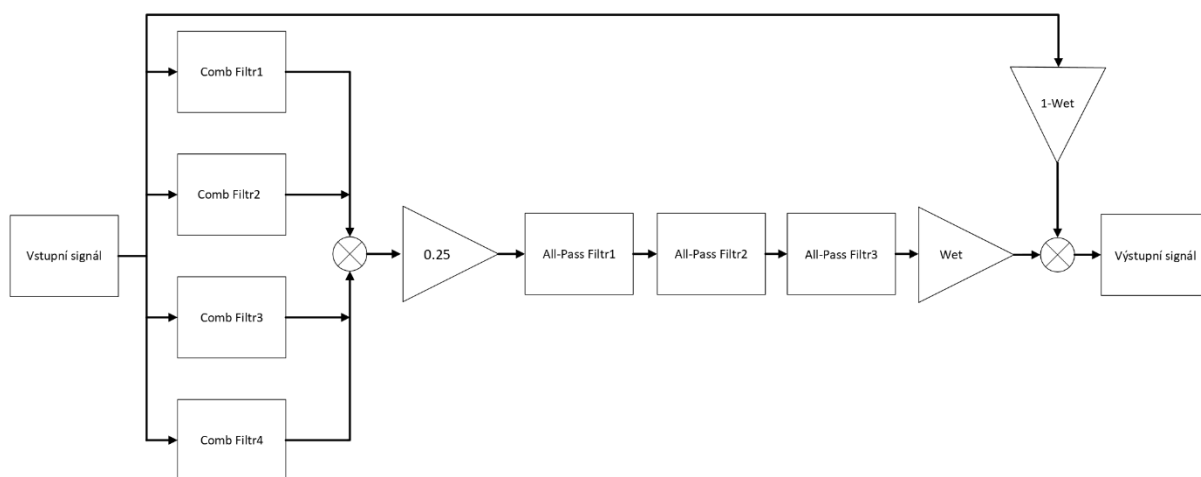
Výstupní RGB data, která představují přímý obraz hudebního signálu, jsou ze STM32 předávána přes další sériové rozhraní USART3 do mikrokontroléru Arduino Uno. To zajišťuje převod těchto dat na standardizovaný formát DMX512, který je poté pravidelně posílán do světelného zařízení Euroliy LC4. Pro lepší vizualizaci je poskytnuto blokové schéma aplikace, které se nachází na Obr. 6.

Součástí celého systému je také digitální modulace zvukového signálu s cílem vytvoření efektu reverbu. Pro realizaci tohoto efektu je využita druhá sestava PMOD I2S2 spolu se samostatným blokem STM32F407G, který zajišťuje implementaci Schroederova reverbu. Tento blok se dělí o stejný vstupní signál s prvním blokem s tím rozdílem, že tento blok má aktivní audio výstup, kde si uživatel pomocí tlačítka volí mezi vstupním a modulovaným signálem.

Celkový návrh systému je optimalizován na minimální zpoždění, stabilitu a spolehlivost v reálném čase. Důraz je kladen na modulárnost řešení, která umožňuje snadné budoucí rozšiřování o další presety, úpravy mapování frekvencí na barvy nebo změnu cílových světelných zařízení.



Obr. 6.: Blokové schéma systému pro ovládání světel pomocí protokolu DMX512



Obr. 7.: Blokové schéma Schroederova reverbu

2.2 Popis použitých komponent

Pro správnou funkci modulační a řídicí jednotky bylo nezbytné zvolit vhodné hardwarové komponenty, které zajistí spolehlivé zpracování audiosignálu, řízení světelné techniky a stabilní napájení celého systému. Výběr jednotlivých součástí byl proveden s ohledem na požadovanou přesnost, rychlost zpracování, kompatibilitu s komunikačními protokoly popsány v teoretické části a dostupnost na trhu. Tato kapitola podrobně popisuje všechny hlavní i podpůrné komponenty, které byly při realizaci zařízení použity.

2.2.1 PMOD I2S2

Pro převod analogového zvukového signálu do digitální podoby byl použit modul PMOD I2S2 (na Obr. 8) od společnosti Digilent. Tento modul obsahuje integrovaný stereo audio převodník Cirrus Logic CS5343, který umožňuje kvalitní digitalizaci audiosignálu s rozlišením 24 bitů a vzorkovací frekvencí až 96 kHz. Modul poskytuje rozhraní I2S pro snadné připojení k mikrokontroléru a minimalizaci rušení přenášeného signálu.

Použití modulu PMOD I2S2 přineslo několik výhod oproti využití integrovaného ADC převodníku přímo v mikrokontroléru STM32F407G. Především šlo o vyšší rozlišení a věrnější přenos zvukového signálu, což je důležité jak pro přesnou frekvenční analýzu, tak pro kvalitní tvorbu efektu reverbu. Modul rovněž umožnil snazší oddělení analogové a digitální části systému, což přispělo ke zvýšení celkové stability a snížení šumu v měřeném signálu.

Pro každý z mikrokontrolérů STM32F407G byl použit samostatný PMOD I2S2 modul. Tím bylo dosaženo oddělení cesty pro analýzu signálu a cesty pro tvorbu zvukového efektu, což napomáhá minimalizaci vzájemného ovlivnění a zajišťuje vyšší kvalitu výsledného



Obr. 8.: PMOD I2S2 [zdroj: rs-online.com]

2.2.2 Obvody CD74HC126

Pro rozdělení digitálních signálů mezi dva moduly PMOD I2S2 byly v projektu použity čtyři logické obvody CD74HC126. Tyto obvody slouží jako nezávislá třístavová hradla, která umožňují duplikaci jednoho digitálního signálu na více výstupů bez vzájemného ovlivňování.

Použití CD74HC126 bylo nezbytné kvůli potřebě rozdělit hodinové a řídicí z mikrokontroléru STM32F407G a jejich připojení na vstupní A/D a D/A části PMOD I2S2 modulu. Každá z těchto částí modulu vyžaduje vlastní vstup hodinového a datového signálu, přičemž přímé paralelní propojení bez použití hradel by mohlo vést ke snížení spolehlivosti nebo poruchám přenosu.

Obvody CD74HC126 zajistily správné rozdělení signálů při zachování kvality a integrity dat. Výstupy hradel byly v projektu nastaveny jako trvale aktivní a napájeny stejným napětím jako ostatní digitální část systému, což zajistilo kompatibilitu úrovní logických signálů.

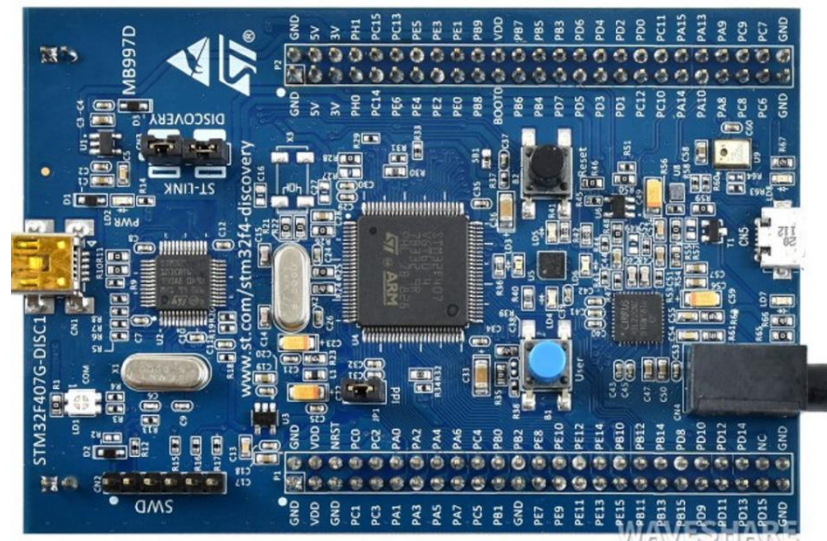
2.2.3 STM32F407G-DISC1

Pro zajištění zpracování audiosignálu, řízení světelné techniky a vytvoření zvukového efektu reverb byly použity dvě vývojové desky STM32F407G-DISC1 (na Obr. 9). Tyto desky obsahují mikrokontrolér STM32F407VG, který je založen na jádru ARM Cortex-M4 běžícím na frekvenci až 168 MHz. Mikrokontrolér je vybaven jednotkou pro digitální zpracování signálu (DSP) a podporou operací s plovoucí desetinnou čárkou (FPU), což je důležité pro rychlé a přesné výpočty. Dále nabízí široké možnosti připojení díky rozhraním USART, I2S a SPI.

Použití dvou vývojových desek místo jedné umožnilo rozdělit náročné úlohy mezi dva nezávislé procesory. První STM32F407G zajišťuje analýzu frekvenčního spektra vstupního audiosignálu a ovládání světelného zařízení pomocí DMX512. Druhý STM32F407G je vyhrazen výhradně pro zpracování zvuku a tvorbu efektu Schroederova reverbu. Toto řešení umožnilo optimalizovat výkon systému, snížit zpoždění při zpracování signálu a zjednodušit ladění obou částí systému.

Původní zadání práce předpokládalo použití vývojové desky STM32H750B-DK, která je postavena na výkonnějším jádru ARM Cortex-M7. Přestože tento mikrokontrolér nabízí vyšší rychlost a větší paměť RAM, jeho použití by přineslo určité komplikace. V době návrhu projektu se vyskytovali potíže s dostupnými knihovnamy STM32CubeIDE pro obousměrnou

komunikaci přes rozhraní I2S, což by zkomplikovalo přenos audiosignálu. Navíc programování tohoto typu mikrokontroléru vyžaduje práci s externí pamětí RAM a složitější konfiguraci systému. Vzhledem k těmto okolnostem bylo rozhodnuto využít dvojici ověřených a stabilně podporovaných desek STM32F407G-DISC1, které plně splňují požadavky práce a umožnily rychlejší a spolehlivější vývoj.



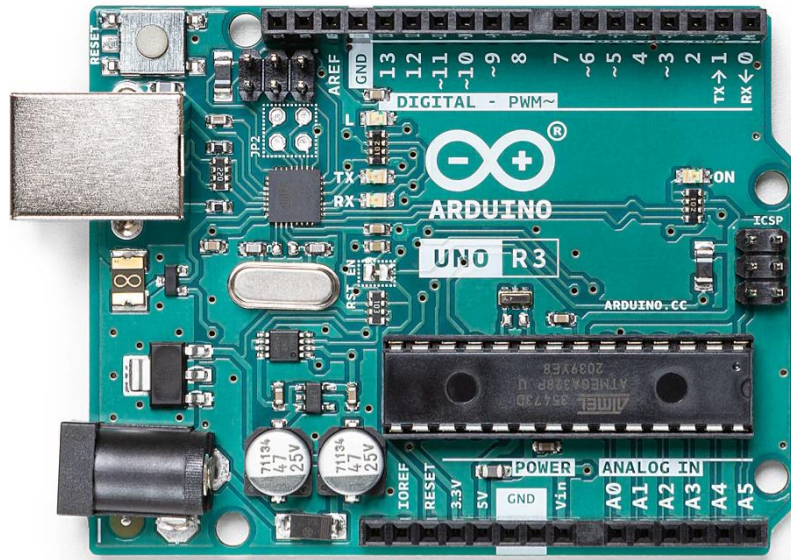
Obr. 9.: STM32F407G-DISC1 [zdroj: dnatechindia.com]

2.2.4 Arduino UNO R3

Pro realizaci komunikace s osvětlovací technikou prostřednictvím protokolu DMX512 bylo v projektu použito vývojové prostředí Arduino Uno R3 (na Obr. 10). Arduino je v tomto případě zodpovědné za příjem dat ve formátu RGB z mikrokontroléru STM32F407G a jejich následné převod do standardního DMX512 signálu, který je odeslán do světelného zařízení.

Použití Arduino Uno bylo zvoleno na základě praktických zkušeností při vývoji. Ačkoli původně bylo v plánu zajistit DMX komunikaci přímo pomocí STM32F407G, v průběhu implementace se objevily technické komplikace. Hlavním problémem byla odlišná logická úroveň signálů – STM32F407G pracuje s 3,3 V logikou, zatímco některá DMX zařízení vyžadují 5V úroveň, což mohlo způsobovat problémy při přímém připojení k RS485 sběrnici. Další komplikací byla skutečnost, že implementace DMX512 obvykle využívá klasické UART rozhraní. Vzhledem k tomu, že STM32F407G v projektu současně používal SPI a I2S rozhraní pro práci s audiem, bylo složité správně nastavit a synchronizovat hodinové signály sdílené mezi více periferiemi.

Arduino Uno R3 umožnilo jednoduché řešení těchto problémů. Pro generování DMX signálu je využíván digitální pin společně s knihovnou DmxSimple.h, která umožňuje softwarové řízení komunikace bez potřeby hardwarového UART. Díky tomu bylo možné vytvořit spolehlivý DMX výstup bez zásahu do ostatních funkcí systému.



Obr. 10.: ArduinoUno R3 [zdroj: arduino.cc]

2.2.5 Převodník MAX485

Pro fyzickou realizaci DMX komunikace byl použit diferenciální transceiver MAX485. Jeho ukázková fotografie se nachází na Obr. 11. Tento obvod převádí signál generovaný digitálním pinem Arduina Uno na diferenciální signál kompatibilní se standardem RS485, který je základem protokolu DMX512. MAX485 je určen pro nízkopříkonovou komunikaci po dlouhých vzdálenostech a umožňuje stabilní přenos dat i v prostředí s vyšším rušením. Je napájen 5V, což je plně kompatibilní s logickými úrovněmi Arduina. Díky jednoduchému zapojení a spolehlivému přenosu byl ideální volbou pro převod signálu na rozhraní DMX512. Výstup z převodníku je propojen se standardním XLR konektorem, kterým je signál dále veden do světelné techniky.



Obr. 11.: Převodník MAX485 [zdroj: laskakit.cz]

2.2.6 Osvětlovací zařízení Eurolite LC4

Pro vizuální demonstraci řízení světelných efektů prostřednictvím protokolu DMX512 bylo v projektu použito zařízení Eurolite LC4 (Obr. 12). Tento světelný efekt obsahuje čtyři výstupy pro RGB LED pásy a umožňuje ovládání barev a intenzity jednotlivých kanálů na základě přijímaných DMX dat.

Eurolite LC4 je kompatibilní se standardem DMX512 a umožňuje jednoduché nastavení startovací DMX adresy, díky čemuž lze snadno definovat, které kanály ovlivňují jednotlivé barvy světla. Ovládání světel bylo realizováno pomocí Arduino Uno, které převádělo barevné hodnoty generované STM32F407G na odpovídající DMX signál odesílaný do zařízení.

Použití Eurolite LC4 bylo zvoleno především kvůli jeho jednoduchosti, spolehlivosti a snadné integraci do systému. Zařízení poskytovalo dostatečnou flexibilitu pro testování různých světelných režimů a umožnilo praktickou demonstraci propojení audio signálu se světelnými efekty.



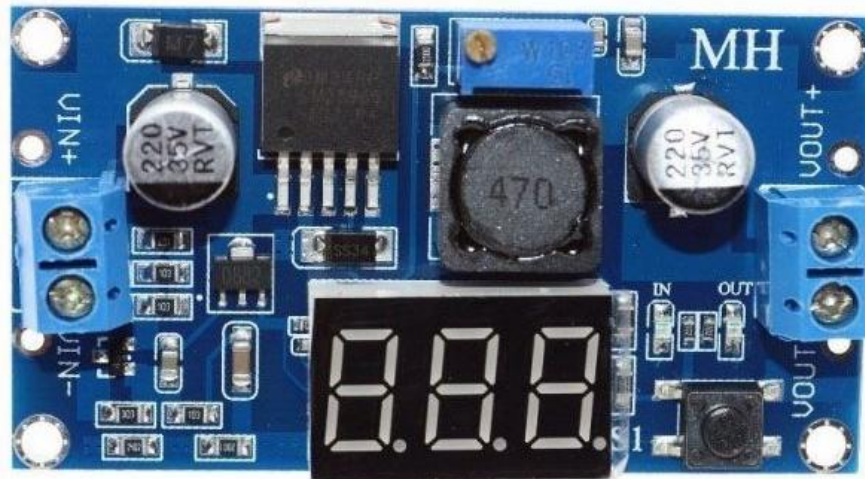
Obr. 12.: Eurolite LC4 [zdroj: Thomann.de]

2.2.3 Napájecí modul LM2596

K napájení celého systému byl použit jeden společný 12V síťový adaptér. Napětí z adaptéru je dále upravováno pomocí dvou samostatných step-down měničů na bázi integrovaného obvodu LM2596 (Obr. 13). Tyto měniče zajišťují stabilní napětí 5 V pro napájení modulů STM32F407G a 7 V pro napájení desky Arduino Uno.

Moduly LM2596 byly zvoleny především díky své jednoduchosti, spolehlivosti a schopnosti efektivně převádět vyšší napětí na požadovanou hodnotu při minimálních ztrátách. Díky použití spínaných měničů je zajištěna vyšší účinnost napájení, což je důležité zejména při napájení více zařízení současně.

Použití oddělených měničů pro 5 V a 7 V napájecí větve umožnilo přesně nastavit požadované napětí pro jednotlivé části systému a minimalizovat vzájemné ovlivnění mezi zdroji. Stabilní napájení je zásadní pro bezproblémový provoz všech elektronických součástí a přispívá k celkové spolehlivosti zařízení.



Obr. 13.: Napájecí step-down modul LM2596 [zdroj: laskakit.cz]

2.2.6 Převodník FT232RL

Pro komunikaci mezi mikrokontrolérem STM32F407G a počítačem byl použit USB-UART převodník založený na obvodu FT232RL (Obr. 14). Tento modul slouží k převodu sériové komunikace USART2 ze STM32 na standardní USB rozhraní, které je možné přímo připojit k běžnému počítači.

Použití převodníku FT232 umožňuje snadné monitorování průběhu frekvenční analýzy a ladění systému v reálném čase. Přes sériové rozhraní je možné zobrazovat výstupní data nebo přepínat mezi jednotlivými provozními režimy. Výhodou modulu FT232 je spolehlivý přenos dat, jednoduchá instalace ovladačů a kompatibilita s většinou operačních systémů bez nutnosti speciálního nastavování. Převodník je propojen se STM32 prostřednictvím rozhraní USART2 a umožňuje obousměrnou komunikaci. Díky tomuto řešení bylo možné snadno vytvořit

uživatelské rozhraní v podobě počítačové aplikace, která s jednotkou komunikuje přes virtuální sériový



Obr. 14.: Převodník FT232RL [zdroj: botland.cz]

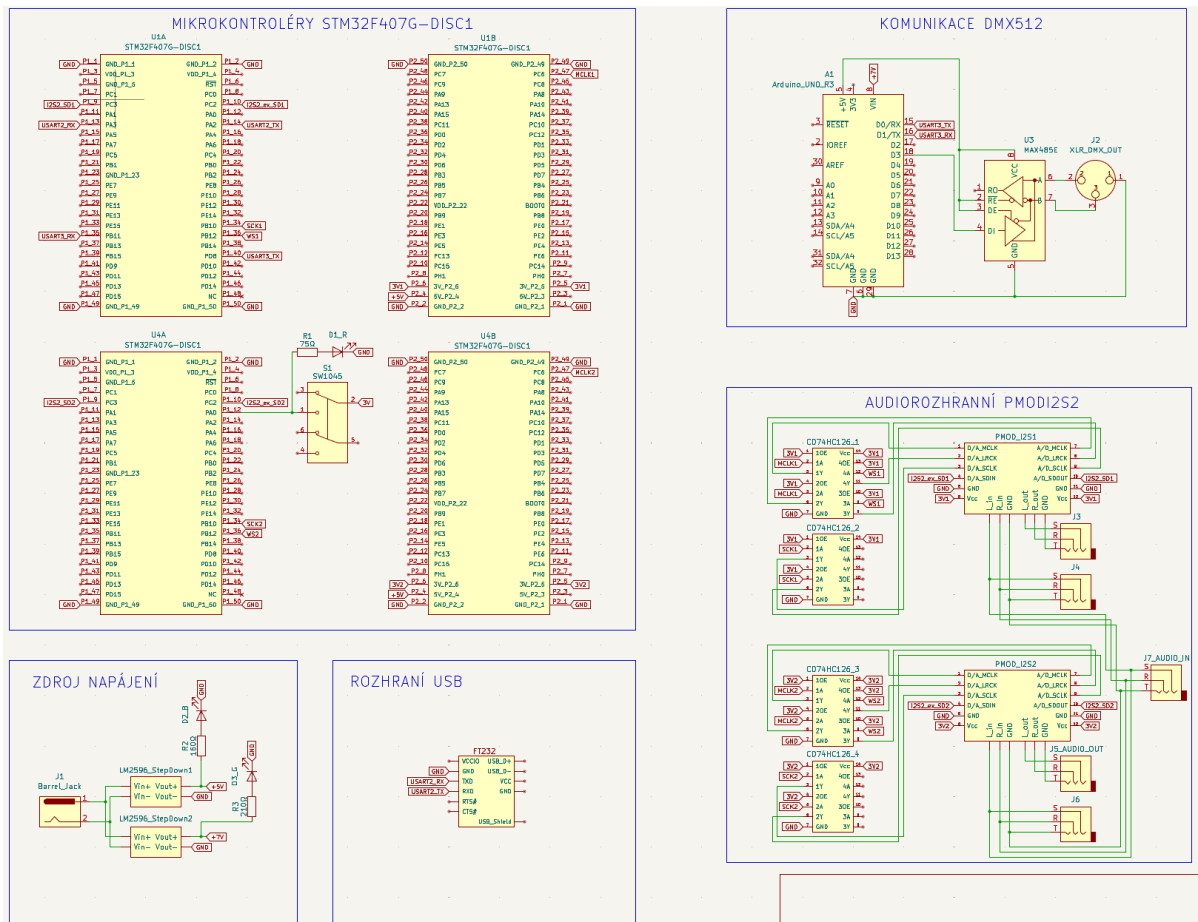
2.3 Realizace hardware

Tato kapitola popisuje praktickou realizaci hardwarové části modulační a řídicí jednotky. Na základě předchozího návrhu systému a výběru vhodných komponent bylo přistoupeno k samotnému fyzickému sestavení zařízení. Cílem této fáze bylo správné propojení jednotlivých částí systému, zajištění stabilního napájení, eliminace nežádoucího rušení a dosažení spolehlivé funkce všech navržených bloků. Popis zahrnuje způsob propojení hlavních modulů, řešení napájecí části, montážní uspořádání a ověření základní funkčnosti po sestavení.

2.3.1 Zapojení hlavních částí systému

Fyzická realizace zařízení vycházela ze schématu zapojení vytvořeného v návrhovém prostředí KiCad. Celý systém je rozdělen na několik samostatných bloků: mikroprocesorové jednotky STM32F407G-DISC1, modul PMOD I2S2 pro digitalizaci audiosignálu, Arduino Uno R3 pro generování DMX512 signálu, převodník MAX485, světelný řídicí modul Eurolite LC4 a napájecí modul na bázi LM2596.

Jednotlivé části byly propojeny podle schématu zapojení, které se nachází na Obr. 15. Datová komunikace mezi moduly byla realizována pomocí standardních vodičů Dupont. Napájení bylo vedeno odděleně pro každou napěťovou úroveň, aby se minimalizovalo rušení a zvýšila stabilita systému.



Obr. 15.: Schéma zapojení hlavní části systému

2.3.2 Fyzické propojení a montáž

Při fyzické realizaci prototypu bylo nejprve nutné propojit jednotlivé součásti systému. Ke spojení mikrokontroléru s dalšími periferiemi byly použity standardní Dupont kabely s dutinkami a kolíky. Tyto propojky umožňují rychlé a opakovatelné zapojení bez potřeby pájení, což usnadňuje ladění systému v průběhu vývoje. Pro připojení komponent osazených do univerzálního pájivého pole byly využity klasické propojovací vodiče, které byly na trvalo připájeny.

Vnější konstrukce zařízení byla navržena s ohledem na praktické používání. Obvodový rám krytu byl vytištěn pomocí 3D tiskárny z PETG plastu, což umožnilo snadné vymodelování otvorů a úchyťů pro jednotlivé konektory. Do této části byly integrovány konektory pro různé typy připojení, konkrétně 6,3mm jack pro audio vstup/výstup, XLR konektor, USB konektor typu A pro komunikaci a konektor typu barrel jack pro napájení. Díky možnosti přesného

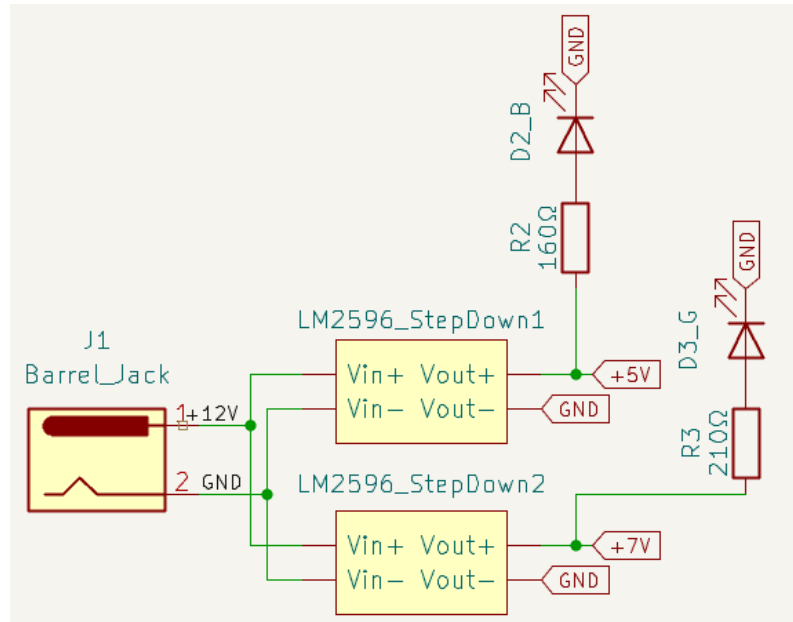
přizpůsobení tvaru a rozměrů 3D tiskem bylo možné dosáhnout pevného uchycení konektorů a zároveň zachovat estetický vzhled zařízení.

Horní a spodní část konstrukce byly vyrobeny z dřevěné překližky, která zajišťuje dostatečnou pevnost a zároveň působí příjemným dojmem při fyzickém kontaktu. Dřevo bylo řezáno na míru a připevněno k tištěnému plastovému rámu pomocí vrtů. Celkově byla montáž navržena tak, aby bylo možné jednotlivé části snadno demontovat a v případě potřeby upravit či vyměnit.

2.3.3 Napájení systému

Napájení celého systému bylo řešeno pomocí jednoho 12V síťového adaptéru. Tento adaptér byl připojen k modulu LM2596, kde bylo napětí pomocí dvou samostatných měničů upraveno na 5 V pro STM32 a 7 V pro Arduino napájené přes VIN pin.

Rozdělení napájecích větví bylo důležité pro stabilní a bezporuchový provoz všech částí systému. Schéma zapojení se nachází na Obr. 16.



Obr. 16.: Schéma zapojení napájecí části systému

2.3.4 Testování základní funkčnosti

Po dokončení fyzického propojení byla testována základní funkčnost systému. Nejprve bylo ověřeno napájení jednotlivých modulů a správná komunikace mezi STM32 a PMOD I2S2. Následně byla testována sériová komunikace mezi STM32 a Arduino a správná interpretace RGB dat v rámci DMX512 protokolu.

Testování světelného výstupu probíhalo postupným generováním konstantní frekvence a kontrolou odpovídající odezvy světelného zařízení Eurolite LC4. Zároveň byla ověřena funkce Schroederova reverbu na výstupním audiokanálu.

Při testování byl kladen důraz na stabilitu přenosu, minimální zpoždění systému a na bezchybnou synchronizaci světelných efektů s audio vstupem.

2.4 Realizace software

Softwarová část projektu byla navržena tak, aby umožňovala zachytávání a zpracování audio signálu v reálném čase, generování efektů a řízení světel pomocí protokolu DMX512. Po spuštění programu dochází k inicializaci všech potřebných periférií mikrokontroléru, zejména rozhraní I2S pro příjem audio dat, rozhraní USART pro komunikaci se světelným systémem a s PC aplikací, a ADC převodníku.

Získaný audio signál je pomocí rozhraní I2S a přímého přístupu do paměti (DMA) ukládán do vyrovnávacích pamětí, odkud je dále zpracováván. Z těchto dat je poté pomocí rychlé Fourierovy transformace (FFT) získáváno spektrální rozložení signálu. Výpočet FFT je prováděn pomocí knihovny ARM CMSIS-DSP, která poskytuje optimalizované matematické funkce pro architekturu Cortex-M4.

Na základě výsledků FFT je provedena analýza spektra a vyhledání dominantní frekvence, tedy té frekvenční složky, která má nejvyšší energii. Současně je vypočítána intenzita této složky, která později určuje jas výsledného světelného efektu.

Dominantní frekvence je následně převedena na odpovídající barvu ve formátu RGB. K tomuto účelu slouží předdefinované mapy, tzv. presety, které přiřazují jednotlivým frekvenčním oblastem konkrétní barevné odstíny. Intenzita frekvence pak přímo ovlivňuje jas výsledné barvy.

Výsledné barevné hodnoty (červená, zelená a modrá složka) jsou zabaleny do 8bitového datového rámce a odeslány prostřednictvím rozhraní UART3 na Arduino Uno, které provádí konverzi těchto dat na DMX signál pro řízení světelného řetězce.

Součástí softwarového řešení je rovněž implementace algoritmického dozvuku (reverbu) podle Schroederova algoritmu. Tento reverb je aplikován přímo na audio signál a dodává mu prostorový charakter.

Kromě samotného zpracování signálu umožňuje program i vzdálené ovládání systému pomocí PC aplikace. Uživatel tak může v reálném čase monitorovat spektrální data a přepínat mezi různými barevnými presety, čímž přizpůsobí světelný výstup aktuálním požadavkům.

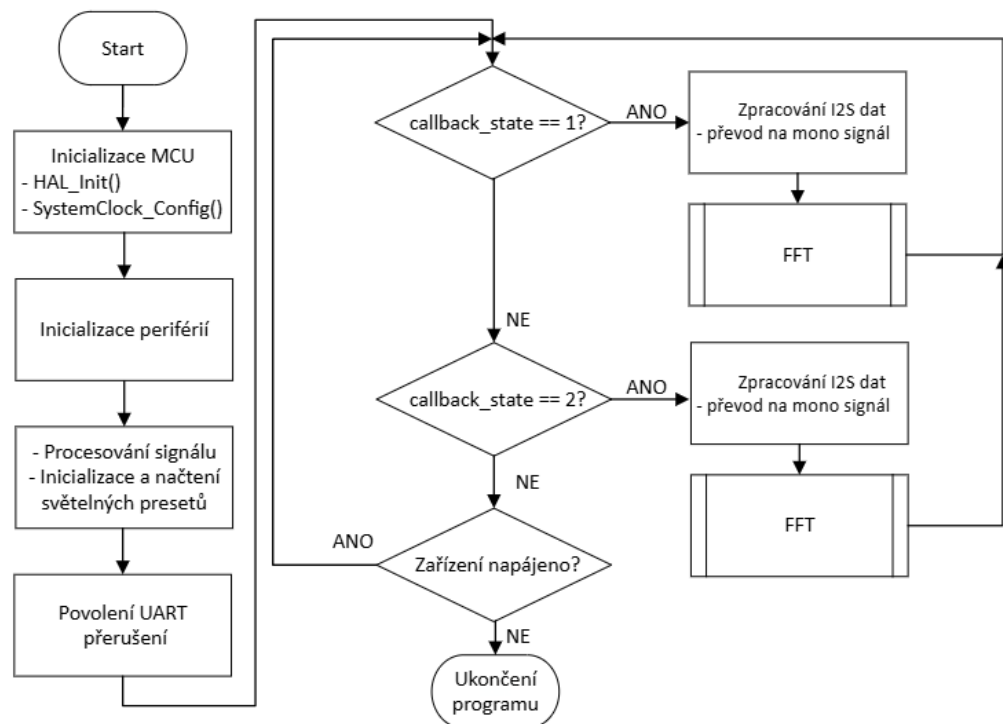
Celý systém byl vyvíjen s důrazem na minimální odezvu, stabilitu a jednoduchou rozšiřitelnost.

2.4.1 Inicializace periférií pro ovládání světel

Hlavní roli v systému sběru audio dat sehrálo rozhraní I2S2. To bylo nakonfigurováno v režimu plného duplexu, což umožnilo oboustranný přenos dat. Vzorkovací frekvence byla nastavena na 48 kHz, což zajistilo dostatečné rozlišení pro následnou spektrální analýzu. Datový formát byl nastaven na 24 bitů, což odpovídá formátu audio vstupu, a bylo aktivováno generování MCLK signálu potřebného pro synchronizaci. Přenos dat byl řešen pomocí rozhraní DMA, které umožnilo rychlý a nepřetržitý přenos velkého množství dat bez zátěže hlavního procesoru.

Pro komunikaci s okolními zařízeními byla použita dvě sériová rozhraní. Rozhraní USART2 sloužilo k přenosu spektrálních dat a přijímání příkazů od PC aplikace. Spektrální data byla odesílána po každém provedení FFT a zároveň bylo možné přijímat příkazy k přepnutí barevného presetu. Pro zvýšení efektivity přenosu byla i zde využita podpora DMA. Druhé rozhraní, USART3, bylo využito pro přímé odesílání barevných hodnot ve formátu RGB na Arduino Uno, které dále zajišťovalo převedení těchto dat na standard DMX512 pro řízení světelného řetězce.

Veškeré inicializační funkce, které jsou součástí hlavního programu na Obr. 17 byly implementovány pomocí knihovny STM32 HAL, která poskytla potřebné abstrakce pro konfiguraci jednotlivých periférií a umožnila snadné propojení hardwaru se zbytkem softwarového řešení.



Obr. 17.: Vývojový diagram hlavního programu STM32 pro ovládání světel

2.4.2 FFT analýza a výběr dominantní frekvence

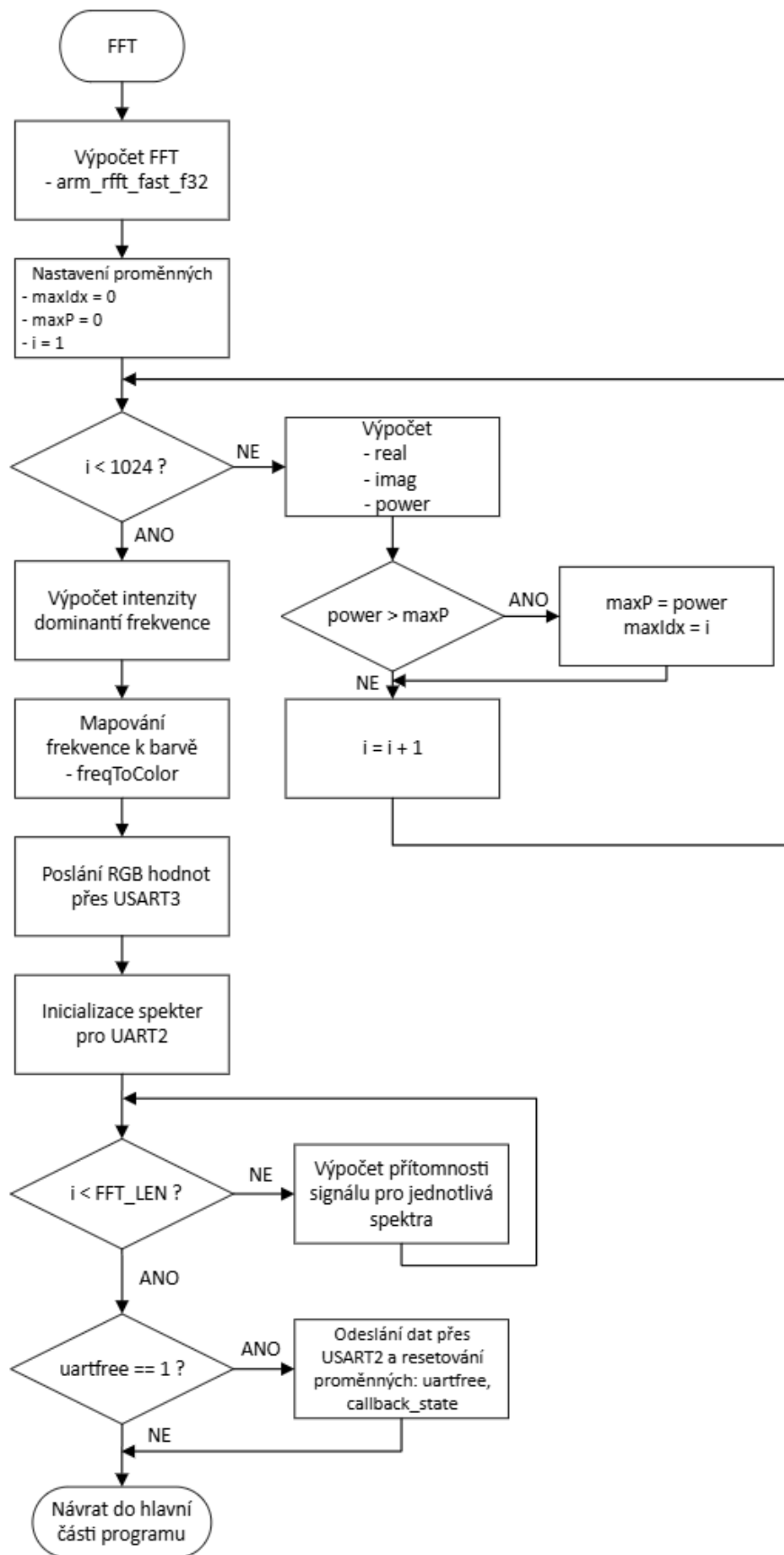
Po načtení audio signálu do vyrovnávací paměti je nutné provést jeho převod ze časové domény do frekvenční domény. K tomuto účelu byla využita rychlá Fourierova transformace (FFT), která umožňuje zjistit spektrální složení vstupního signálu. Implementace byla provedena pomocí optimalizované knihovny ARM CMSIS-DSP pro jádro Cortex-M4, konkrétně funkcí `arm_rfft_fast_f32()`.

Data ze vstupní paměti `rxBuf` jsou před transformací převedena do pole `fft_in_buf` ve formátu plovoucí desetinné čárky. Vzhledem k použití rozhraní I2S v plném duplexu je nutné sloučit dvojice vzorků do jednoho bodu, přičemž se provádí jednoduché součtování dvou kanálů na jeden mono signál. Tento krok je realizován v hlavní smyčce programu, jakmile je přijat dostatek vzorků prostřednictvím přerušení od DMA.

Samotná funkce `FFT()` provádí transformaci vstupních dat a následné vyhodnocení spektrálního maxima. Během průchodu spektrem je v každém kroku vypočten výkon (power) dané frekvenční složky jako součet druhých mocnin reálné a imaginární části. Program následně hledá maximální hodnotu výkonu a její příslušný index, což reprezentuje dominantní frekvenci v signálu.

Po nalezení dominantní frekvence je vypočtena její intenzita v decibelech pomocí logaritmické stupnice. Tato intenzita je dále omezena na rozsah 0 až 255, aby odpovídala rozsahu jasu barevných složek RGB. Dominantní frekvenční index je následně použit pro výběr příslušné barvy z aktivního presetu a výsledná barevná složka je odeslána prostřednictvím rozhraní UART na světelný systém.

Schéma tohoto procesu je znázorněno na Obr. 18. Praktická implementace této funkce je uvedena v následující ukázce kódu na **Chyba! Nenalezen zdroj odkazů..**



Obr. 18.: Vývojový diagram funkce FFT

```

void FFT() {
    arm_rfft_fast_f32(&fft_handler, fft_in_buf, fft_out_buf, 0);

    int maxIdx = 0;
    float maxP = 0;

    for (int i = 1; i < 1024; i++) {
        float real = fft_out_buf[2*i];
        float imag = fft_out_buf[2*i+1];
        float power = real * real + imag * imag;
        if (power > maxP) {
            maxP = power;
            maxIdx = i;
        }
    }

    int inten = (int)(10.0f * log10f(maxP + 1e-6f)) - 150;
    if (inten < 0) inten = 0;
    if (inten > 255) inten = 255;

    freqToColor(maxIdx, inten);

    uint8_t usartData[4] = {0xFF, red, green, blue};
    HAL_UART_Transmit(&huart3, usartData, 4, 100);

    int freqs[1024];
    int freqpoint = 0;
    int offset = 165;
    for (int i = 0; i < FFT_LEN; i += 2) {
        freqs[freqpoint] = (int)(20 * log10f(complexABS(fft_out_buf[i], fft_out_buf[i + 1])) - offset);
        if (freqs[freqpoint] < 0) freqs[freqpoint] = 0;
        if (freqs[freqpoint] > 255) freqs[freqpoint] = 255;
        freqpoint++;
    }

    outarray[0] = 0xff;
    outarray[1] = (uint8_t)freqs[4];
    outarray[2] = (uint8_t)freqs[7];
    outarray[3] = (uint8_t)freqs[12];
    outarray[4] = (uint8_t)freqs[19];
    outarray[5] = (uint8_t)freqs[24];
    outarray[6] = (uint8_t)freqs[30];
    outarray[7] = (uint8_t)freqs[36];
    outarray[8] = (uint8_t)freqs[42];
    outarray[9] = (uint8_t)freqs[48];
    outarray[10] = (uint8_t)freqs[57];

    if (uartfree == 1) {
        HAL_UART_Transmit_DMA(&huart2, outarray, 11);
        uartfree = 0;
        callback_state = 0;
    }
}

```

Obr. 19.: Ukázka výpisu funkce FFT

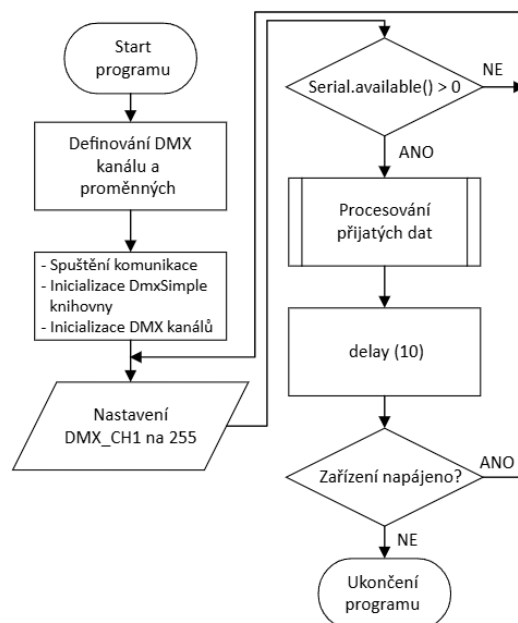
2.4.3 Ovládání světél – Arduino přijímač

Pro převod přijatých barevných dat ze STM32 na výstupní DMX signál byl využit mikrokontrolér Arduino Uno. Arduino obdrželo po sériovém rozhraní (UART) datové rámce z STM32, dekodovalo jejich obsah a prostřednictvím knihovny DmxSimple nastavovalo hodnoty na příslušných DMX kanálech. Tím byl řízen světelný efekt na základě barvy vyhodnocené z dominantní frekvence ve zvukovém signálu.

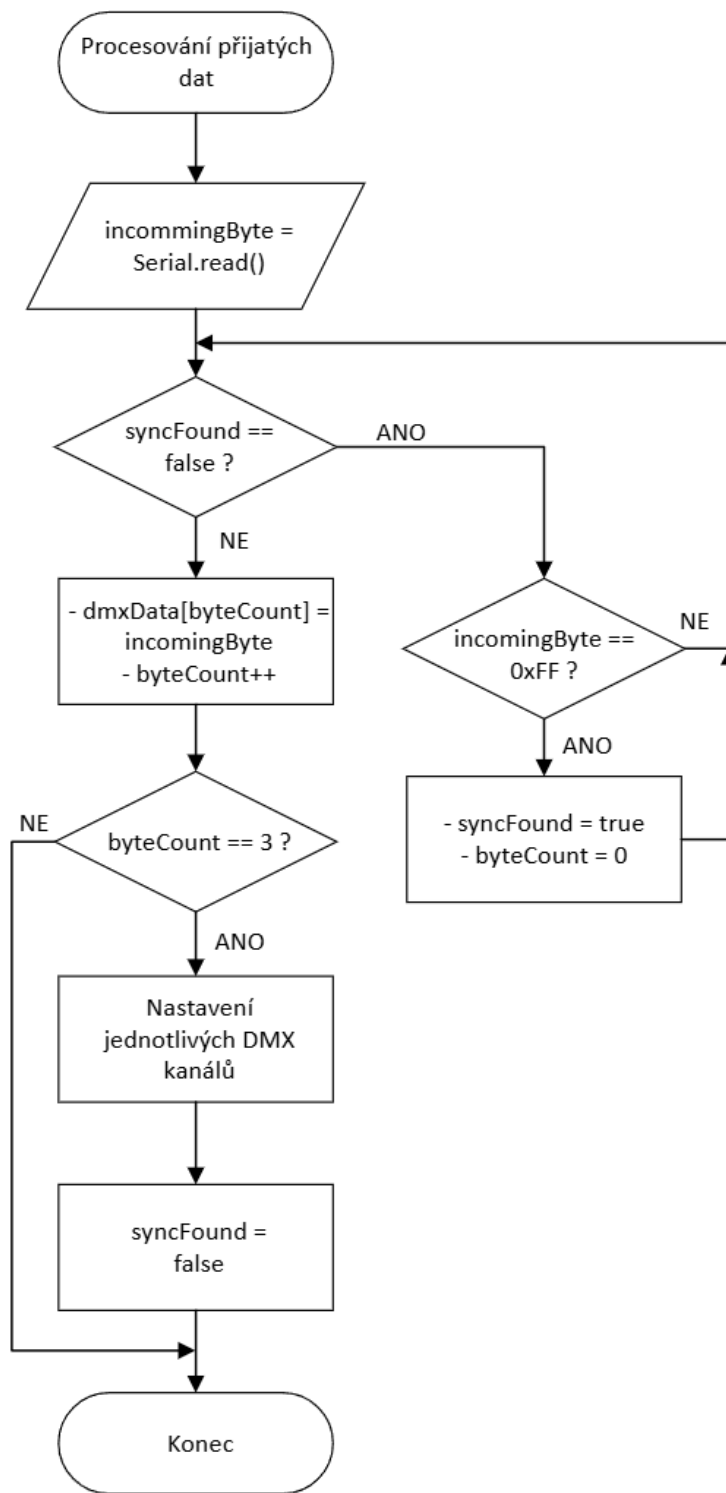
Knihovna DmxSimple je navržena tak, aby co nejvíce připomínala práci se standardními výstupy například každému kanálu lze přiřadit hodnotu pomocí příkazu `DmxSimple.write(kanál, hodnota)`. Funkce `DmxSimple.usePin(pin)` slouží k nastavení výstupního pinu, na který je připojen fyzický DMX vysílač. Knihovna běží nezávisle na hlavní smyčce programu a zajišťuje nepřetržité vysílání datového rámce s aktuálními hodnotami kanálů.

Přenos dat probíhá v rámci o čtyřech bajtech, kde první bajt je vždy synchronizační (0xFF), a následují tři bajty určující jednotlivé barevné složky (červená, zelená a modrá). Po detekci synchronizačního bajtu Arduino postupně načte tři další hodnoty a ty uloží do pole `dmxData[]`. Tyto hodnoty jsou následně odeslány na DMX kanály 2, 3 a 4.

Na kanálu 1 je pevně nastavena hodnota 255, která pro Eurolite LC4 slouží k výběru provozního módu. Použitá knihovna DmxSimple zajišťuje základní komunikaci po DMX sběrnici s využitím vyhrazeného digitálního výstupu. Níže jsou uvedeny vývojové diagramy Arduino kódu na Obr. 20 a Obr. 21.



Obr. 20.: Vývojový diagram hlavního programu pro Arduino Uno



Obr. 21.: Vývojový diagram procesování přijatých dat pro Arduino

2.4.4 Aplikace pro monitorování a ovládání presetů

Pro účely monitorování spektrálních dat a ovládání světelných presetů byla vyvinuta jednoduchá aplikace v jazyce C# s využitím prostředí Windows Forms. Hlavní funkcí této aplikace bylo přijímat data ze STM32 přes sériovou linku, graficky zobrazovat intenzity jednotlivých frekvenčních pásem pomocí progress barů a umožnit uživateli vzdáleně přepínat barevné presety.

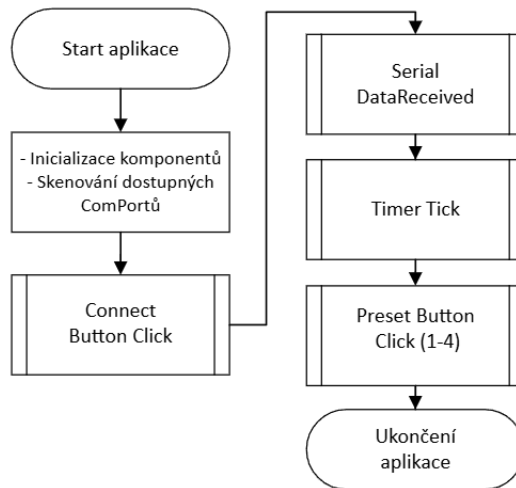
Po spuštění programu dojde k vyhledání všech dostupných sériových portů připojených k počítači. Tyto porty jsou následně zobrazeny v seznamu, ze kterého si uživatel může vybrat ten správný. Po výběru portu a stisku tlačítka Connect se aplikace pokusí navázat spojení s mikrokontrolérem STM32. Pokud je připojení úspěšné, dojde k inicializaci objektu SerialPort, nastavení parametrů komunikace. Na tento port je následně navázána událost DataReceived, která zpracovává veškerá přijatá data ze zařízení.

Přijatá data jsou zpracovávána ve funkci port_DataReceived(). Po příjmu dat ze sériového portu program nejprve ověřuje, zda rámec obsahuje synchronizační bajt 0xFF. Tento bajt slouží k identifikaci začátku platného datového rámce, který obsahuje deset hodnot reprezentujících intenzity vybraných frekvenčních pásem. Data jsou dočasně ukládána do kruhového bufferu, ze kterého se poté správně sestaví pole freqarray. Toto pole slouží jako zdroj dat pro grafické zobrazení v uživatelském rozhraní.

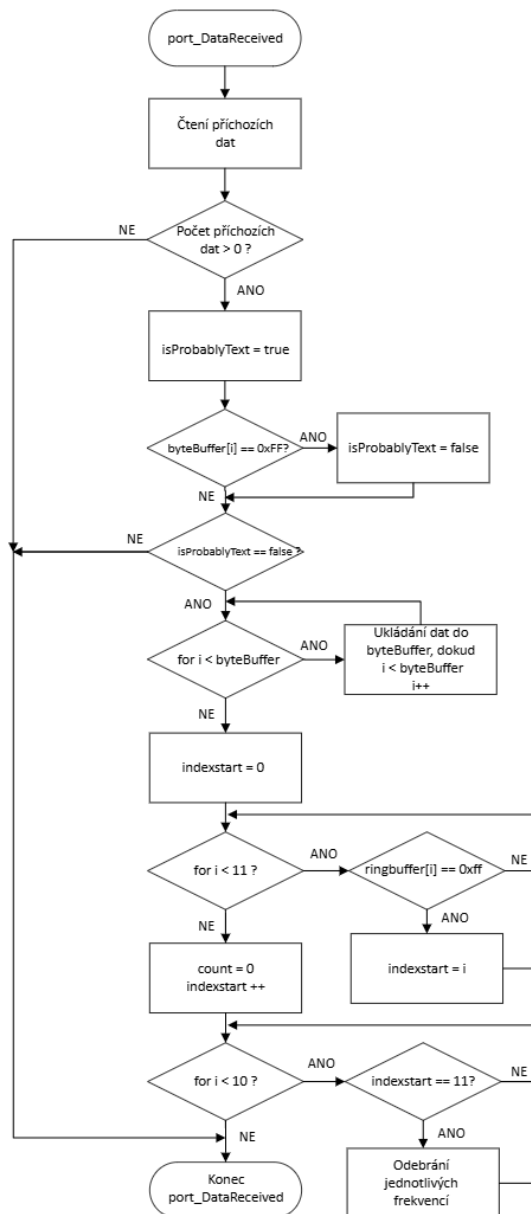
Grafická aktualizace aplikace probíhá v pravidelných intervalech pomocí časovače, jehož událost timer1_Tick() je vyvolávána každých 50 ms. V této události je každému progress baru přiřazena aktuální hodnota z pole freqarray, čímž se v reálném čase aktualizuje vizuální zobrazení síly jednotlivých frekvenčních pásem.

Uživatel má kromě monitorování také možnost zasahovat do chování systému pomocí čtyř tlačítek Preset1 - 4. Po stisknutí některého z těchto tlačítek je odeslán znak ('1', '2', '3' nebo '4') na sériový port, čímž dojde ke vzdálenému přepnutí aktivního barevného presetu přímo na zařízení STM32. Odesílání těchto příkazů zajišťuje funkce SendPresetId().

Architektura aplikace je znázorněna na vývojovém diagramu na Obr. 22 a Obr. 233, který shrnuje průběh hlavních operací od navázání spojení až po zpracování a vizualizaci dat. Jednotlivé funkce jsou dále podrobně rozpracovány ve specifických vývojových diagramech, které znázorňují logiku spojení, příjem dat i aktualizaci grafického rozhraní.



Obr. 22.: Vývojový diagram hlavního kódu aplikace



Obr. 23.: Vývojový diagram funkce port_DataReceived

2.4.5 Tvorba reverbu

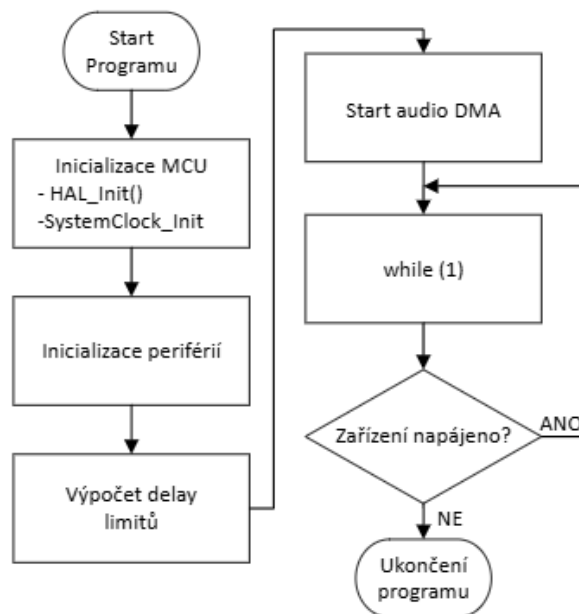
Součástí projektu byla také tvorba samostatné aplikace, která na zpracovaný audio signál přidává reverb. Program běží na mikrokontroléru STM32F407 a využívá rozhraní I2S pro příjem a odesílání zvukového signálu. Přenos dat probíhá v režimu plného duplexu. Aby mikrokontrolér nebyl zbytečně zatížen, je pro přenos použit DMA řadič.

Samotný reverb je tvořen čtyřmi paralelními kombinačními filtry (comb filtry) a třemi sériově zapojenými all-pass filtry. Každý filtr má svoje vlastní buffer, kde si ukládá předchozí vzorky signálu, a pracuje s určitou zpětnou vazbou. Délky jednotlivých filtrů byly upraveny tak, aby odpovídaly vzorkovací frekvenci 96 kHz a zároveň bylo možné všechny buffery uložit do vnitřní paměti STM32.

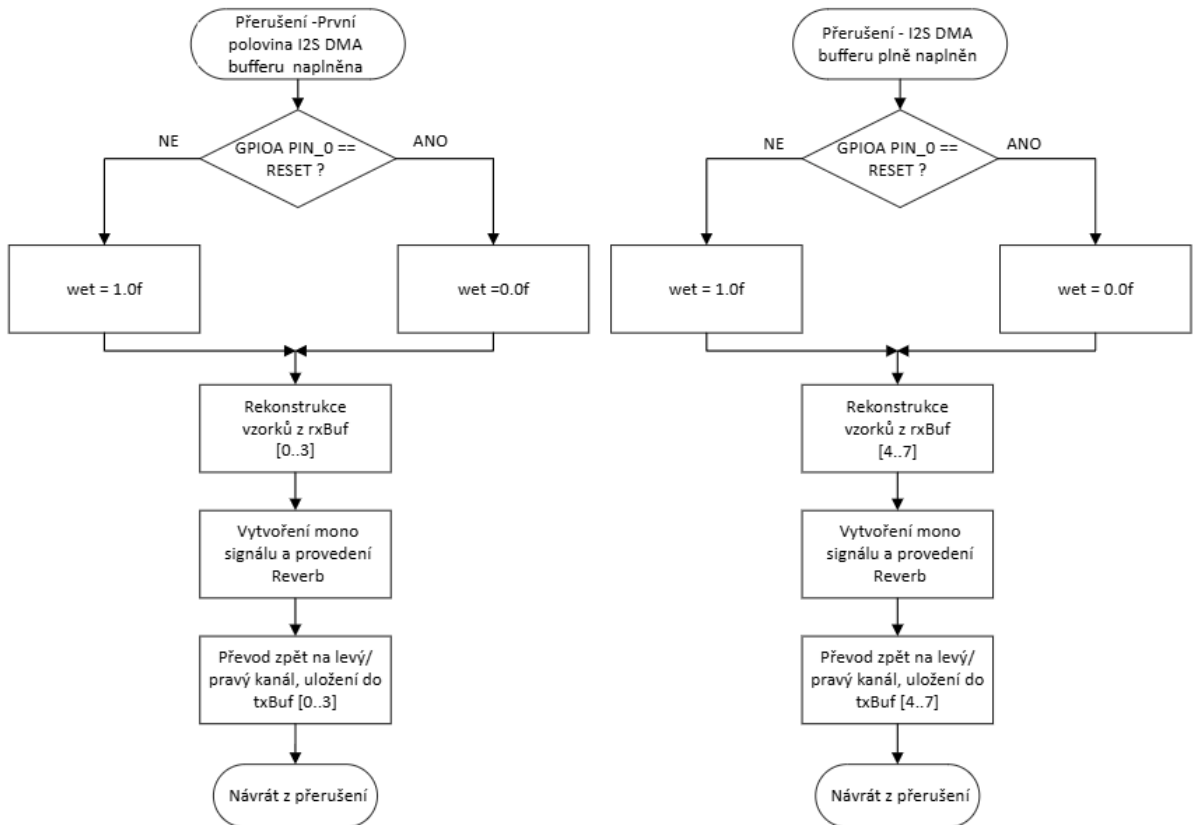
Když mikrokontrolér přijme nový blok dat, nejprve převede přijaté vzorky ze 24bitového formátu na čísla s plovoucí desetinnou čárkou. Tyto vzorky se pak zpracují v reverbovém algoritmu. Výsledek se na výstupu kombinuje s původním signálem, a to podle hodnoty proměnné wet, která určuje, kolik efektu se přimíchá do výstupu.

Úroveň efektu (tedy hodnota wet) je možné ovládat pomocí tlačítka připojeného na pin PA0. Pokud je tlačítko stisknuté, efekt je zapnutý a výstupní signál obsahuje reverb. Pokud je tlačítko uvolněné, přehrává se pouze čistý signál.

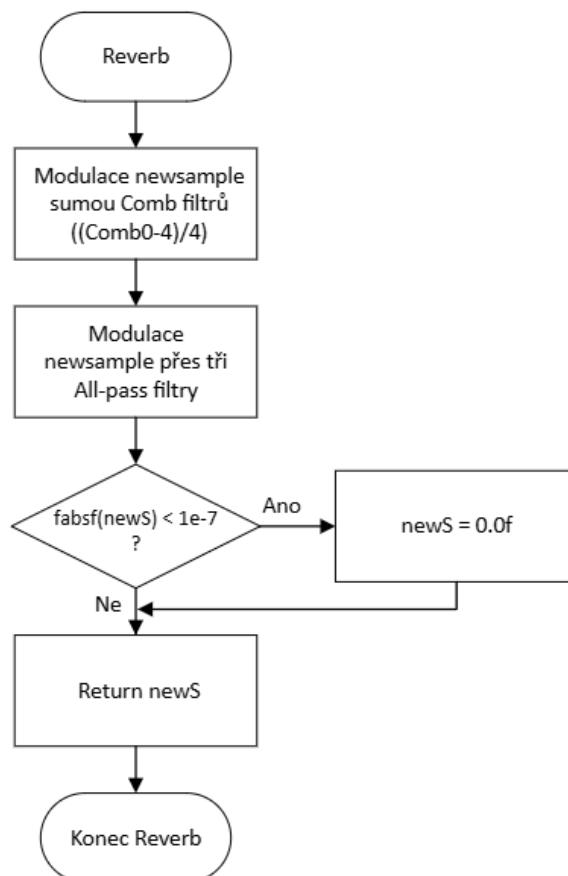
Nakonec se zpracovaný signál převede zpět na 24bitový formát a odešle přes rozhraní I2S ven. Všechno s minimálním zpožděním, takže je možné tento efekt používat i při živém hraní.



Obr. 24.: Hlavní kód programu pro tvorbu reverbu



Obr. 26.: Vývojové diagramy přerušení spuštěného po naplnění I2S DMA bufferu



Obr. 25.: Vývojový diagram funkce Reverb

3 ZÁVĚR

Tato bakalářská práce se zaměřila na návrh a realizaci digitálního systému, který zpracovává signál z elektrické kytary a zároveň na základě tohoto signálu ovládá světelný výstup pomocí protokolu DMX512. Hlavním cílem bylo vytvořit zařízení, které v reálném čase aplikuje zvukové efekty, analyzuje frekvenci a hlasitost vstupu a podle těchto údajů mění barvu a intenzitu světla. Všechny body stanovené v zadání se podařilo úspěšně splnit.

Během práce byl navržen a zprovozněn systém, který dokáže zpracovávat analogový signál z kytary pomocí vestavěného převodníku ADC, aplikovat na něj efekty jako reverb a následně provést frekvenční analýzu pomocí algoritmu FFT. Výsledné údaje o frekvenci a amplitudě jsou odesílány přes sériovou linku a využívány pro ovládání barev a intenzity RGB světla. Všechny části systému, od snímání signálu přes digitální zpracování až po ovládání světla, spolu pracují spolehlivě a v reálném čase.

Projekt ukázal, že i s omezenými prostředky je možné vytvořit funkční zařízení kombinující zvukové efekty, analýzu signálu a řízení periférií. Podařilo se propojit různé oblasti, práci s mikroprocesorem, digitální zpracování signálu a řízení osvětlení, do jednoho funkčního celku.

Součástí řešení byla i návrhová a konstrukční část. Výsledné zařízení bylo osazeno do vlastnoručně zhotovené dřevěno-plastové krabičky, která je navržena s důrazem na mechanickou odolnost a praktické využití v reálném prostředí. Konstrukce obsahuje přehledně rozmístěné konektory pro audio vstup a výstup, DMX výstup, napájení a komunikaci s PC. Na horní straně je dále umístěn nožní spínač a indikační LED diody signalizující stav systému (Obr. 27). Připojený LED bar je rovněž vlastní konstrukce a využívá RGB LED pásek řízený pomocí DMX dekodéru Eurolite LC4, umístěného v samostatné krabici. LED bar lze vidět na Obr. 28.

Součástí projektu je také ovládací aplikace vytvořená ve Visual Studio Forms, která umožňuje monitorování frekvenčních pásem a výběr barevných presetů. Aplikace slouží k pohodlnému testování a ovládání systému.

Praktická realizace prokázala, že navržený systém je plně funkční a použitelný. Přesto existují oblasti, které by bylo možné dále zlepšit. Nejvýraznější z nich je určitá latence pozorovaná při použití efektu reverb. Tato latence je způsobena velikostí bufferů použitých ve filtrech typu Comb a All-pass, které byly záměrně nastaveny tak, aby byl reverb co nejvýraznější. Vzhledem

k omezené paměťové kapacitě použitého mikrokontroléru dochází ke zpoždění v řádu několika milisekund, které může být v některých případech slyšitelné.

Pro snížení této latence by bylo možné v budoucnu zvolit výkonnější hardware s větší pamětí, případně optimalizovat paměťové nároky efektového modulu, například zmenšením velikostí reverb bufferů. Tyto úpravy by mohly zlepšit celkovou odezvu systému.

Celkově lze projekt hodnotit jako úspěšný s potenciálem pro optimalizaci a využití v reálných aplikacích.



Obr. 27.: Vrní pohled na finální konstrukci systému



Obr. 28.: Finální konstrukce DMX LED baru

ZDROJE ELEKTRONICKÉ

NTi Audio. *Fast Fourier Transform (FFT)* [online]. LIESTAL: NTi Audio AG, [cit. 2025-04-30]. Dostupné z: <https://www.nti-audio.com/en/support/know-how/fast-fourier-transform-fft>

DUNN, Robert. *Introduction to the I²S Interface* [online]. All About Circuits, 2020 [cit. 2025-04-30]. Dostupné z: <https://www.allaboutcircuits.com/technical-articles/introduction-to-the-i2s-interface/>

KORBA, Petr. *DMX512 timing diagram (MTBP: Mark Time Between Frames, MAB: Mark After Break)* [online]. In: ResearchGate, 2013 [cit. 2025-04-30]. Dostupné z: https://www.researchgate.net/figure/DMX512-timing-diagram-MTBT-Mark-Time-Between-Frames-MAB-Mark-After-Break_fig2_263633914

DIGILENT. *Pmod I2S2, Analogový vývojový nástroj pro AD/DA převodník, 3.3 V, PMOD* [online]. RS Components, [cit. 2025-04-30]. Dostupné z: <https://cz.rs-online.com/web/p/analogove-vyvojove-nastroje/1840481>

DNA TECHNOLOGY. *STM32F407G-DISC1 Discovery Kit* [online]. [cit. 2025-04-30]. Dostupné z: <https://www.dnatechindia.com/stm-32-f-407g-disc-discovery-kit-buy-in-india.html>

ARDUINO. *Arduino UNO Rev3* [online]. Arduino Store, [cit. 2025-04-30]. Dostupné z: <https://store.arduino.cc/en-cz/products/arduino-uno-rev3>

LASKAKIT. *Převodník TTL na RS-485, MAX485* [online]. [cit. 2025-04-30]. Dostupné z: <https://www.laskakit.cz/prevodnik-ttl-na-rs-485--max485/>

EUROLITE. *LC-4 LED Strip RGB DMX Control* [online]. Thomann GmbH, [cit. 2025-04-30]. Dostupné z: https://www.thomann.de/cz/eurolite_lc_4_led_strip_rgb_dmx_control.htm

LASKAKIT. *Step-down měnič s LM2596 s LED displejem* [online]. [cit. 2025-04-30]. Dostupné z: <https://www.laskakit.cz/step-down-menic-s-lm2596-s-led-displejem/>

WAVESHARE. *Převodník USB-UART FTDI FT232RL, USB zástrčka* [online]. Botland.cz, [cit. 2025-04-30]. Dostupné z: <https://botland.cz/prevadece-usb-uart-rs232-rs485/4502-prevodnik-usb-uart-ftdi-ft232rl-usb-zastrcka-waveshare-6170-5904422374013.html>

SEZNAM PŘÍLOH

Příloha A: Finální vzhled systému

Příloha B: LED bar vyhotovený pro demonstraci projektu

Příloha C: Vzhled ovládací a monitorovací aplikace

Přílohy D: CD

PŘÍLOHA A: Finální vzhled systému

Na horní straně se nachází nožní spínač a tři kontrolní LED diody – zelená signalizuje napájení Arduina Uno, modrá napájení STM32 a červená indikuje aktivaci reverb efektu. Na pravé straně je umístěn XLR konektor pro výstup DMX signálu a dvojice 6,3mm jack konektorů pro vstupní a výstupní audio signál. Levá strana obsahuje napájecí konektor (12 V) a USB port pro připojení k počítači. Konstrukce je pevná, kompaktní a vhodná pro mobilní použití.



PŘÍLOHA B: LED bar vyhotovený pro demonstraci projektu

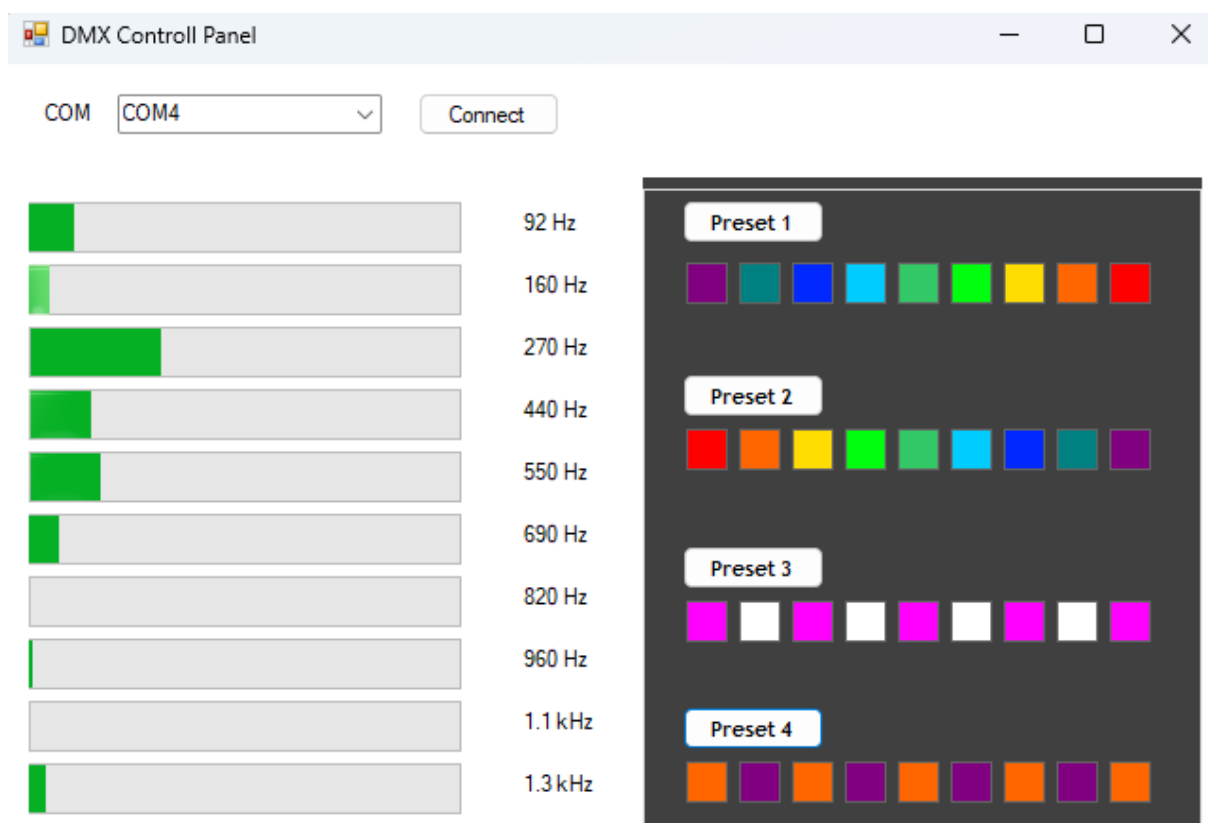
LED bar je umístěn v instalačním boxu, ze kterého vystupuje svislý dřevěný profil s nalepeným RGB LED páskem. Na horní straně se nachází napájecí konektor (12 V) a zelená LED signalizující přítomnost napájení. Pravá strana obsahuje DMX vstup pro řízení z hlavní jednotky. Uvnitř je umístěn DMX dekodér Eurolite LC4, který převádí přijatý signál na ovládání LED pásku.



PŘÍLOHA C: Vzhled ovládací a monitorovací aplikace

Ovládací aplikace byla vytvořena ve vývojovém prostředí Visual Studio pomocí Windows Forms. V horní části okna se nachází rozbalovací nabídka (ComboBox) pro výběr sériového portu a tlačítko „Connect“ pro navázání komunikace se systémem. Levá část rozhraní zobrazuje devět svislých indikátorů (progress barů), které zobrazují úroveň signálu v jednotlivých frekvenčních pásmech.

Pravá část aplikace slouží pro výběr barevných presetů. Každý preset obsahuje devět barevných čtverců, z nichž každý odpovídá jednomu frekvenčnímu pásmu z levé části – řazení je zleva doprava (první čtverec odpovídá hornímu progress baru). Nad každým schématem je umístěno tlačítko, kterým lze daný preset aktivovat.



PŘÍLOHA D

Příloha k bakalářské práci

Modulační a řídicí jednotka světelné a zvukové techniky s rozhraním DMX512

Ondřej Vozáb

CD

OBSAH

1. Text bakalářské práce ve formátu PDF
2. Schémata zapojení
3. Zdrojový kód Arduino Uno
4. Zdrojový kód STM32
5. Zdrojový kód Visual Studio Forms