



Středoškolská technika 2023

Setkání a prezentace prací středoškolských studentů na ČVUT

SAMOHRAJÍCÍ XYLOFON

Tomáš Janoušek

SPŠE V Úžlabině
V Úžlabině 320, Praha 10

Čestné prohlášení

Odevzdáním této maturitní práce na téma **Samohrající Xylofon** potvrzuji, že jsem ji vypracoval/a pod vedením vedoucího samostatně za použití v práci uvedených pramenů a literatury. Dále potvrzuji, že odevzdaná vytištěná verze písemné zprávy (protokolu) a plakátu se plně shoduje s odevzdanou elektronickou verzí.

V Praze dne.....

.....
Podpis autora/autorky práce

Anotace

Tato maturitní práce se věnuje problematice, jak přehrát MIDI soubory na xylofonu, za pomoci mikrokontroleru. Součástí práce také je stvoření mobilní aplikace pro Android, která je schopna do xylofonu posílat MIDI soubory anebo převzít nad xylofonem kontrolu a začít na něj hrát přímo. Důležité je pochopit, jak funguje samotný MIDI protokol a vymyslet způsob, jak ho převést na údery kamenů xylofonu.

Klíčová slova

Bluetooth, ESP32, MIDI, Mobilní aplikace

Annotation

This graduation thesis deals with the issue of how to play MIDI files on a xylophone using a microcontroller. The work also includes the creation of a mobile application for Android, which is able to send MIDI files to the xylophone or take control over the xylophone and start playing it directly. It is important to understand how the MIDI protocol itself works and to figure out a way to convert it to the strikes of xylophone stones.

Keywords

Bluetooth, ESP32, MIDI, Mobile application

Obsah

1	ÚVOD A CÍL PRÁCE	5
2	TEORETICKÁ ČÁST PRÁCE	6
2.1	SMF FORMAT	6
2.1.1	<i>Typy MIDI zpráv</i>	<i>6</i>
2.1.2	<i>Header Chunk.....</i>	<i>6</i>
2.1.3	<i>Track Chunk.....</i>	<i>6</i>
2.2	VÝBĚR ŘEŠENÍ	8
2.3	VÝBĚR A POPIS SOUČÁSTEK.....	8
2.4	VÝPOČET CENY PROJEKTU.....	12
3	PRAKTICKÁ ČÁST PRÁCE	13
3.1	UPEVNĚNÍ SOLENOIDŮ.....	13
3.2	DESKA PLOŠNÉHO SPOJE	13
3.3	OVLÁDACÍ MODUL	14
3.4	ŘÍDÍCÍ PROGRAM	14
3.4.1	<i>Vypisování na obrazovku</i>	<i>14</i>
3.4.2	<i>Čtení dat z microSD karty</i>	<i>14</i>
3.4.3	<i>Implementace parsování.....</i>	<i>14</i>
3.4.4	<i>Přehrávání</i>	<i>15</i>
3.4.5	<i>Bluetooth.....</i>	<i>15</i>
3.5	MOBILNÍ APLIKACE	15
3.6	NAPÁJENÍ VÝROBKU	16
3.7	3D TISK	17
4	ZÁVĚR	18
5	SEZNAM POUŽITÝCH ZDROJŮ.....	19
6	SEZNAM PŘÍLOH.....	20

1 Úvod a cíl práce

Projekt samohrajícího xylofonu je spojením technologie a umění, které umožňuje vytvořit zařízení, jež dokáže hrát hudbu na xylofonu bez přímého zásahu člověka. Umožňuje tak i lidem se sníženou pohyblivostí či zrakově postiženým, pro které by bylo normálně velmi obtížné nebo i nemožné hrát na jakýkoliv hudební nástroj, aby mohly hrát a rozvíjet tak svůj hudební talent.

Samohrající xylofon by měl fungovat na bázi MIDI protokolu, což je standardizovaný způsob, jak digitálně přenášet hudební informace mezi různými hudebními zařízeními. Tyto soubory obsahují informace o notách, rytmu, dynamice a dalších hudebních parametrech, který samohrající xylofon převádí na údery do kamenů xylofonu.

Kromě ovládání přímo na výrobku mělo by také být možné ovládat xylofon bezdrátově pomocí mobilní aplikace, která bude moct posílat data nebo i xylofon ovládat přímo.

V této práci se nejdříve zaměřím na popis MIDI protokolu a řešení předložených problematik a poté na to, jak byl projekt samohrajícího xylofonu vytvořen a jaké technologie a postupy byly použity.

2 Teoretická část práce

V této části se budu věnovat teoretickým řešením předložených problémů a jejich řešení. Hlavními částmi je probrání MIDI specifikace, vybraný způsob řešení a výběr součástek.

2.1 SMF format

Nejdříve je důležité pochopit samotný formát MIDI souborů označovaný jako SMF (Standard MIDI file). V této kapitole se věnuji parametrům SMF a jak dané soubory přečíst.

2.1.1 Typy MIDI zpráv

Buď posílám data bajty nebo status bajty. Data bajty nabývají hexadecimální hodnoty od 0x00 do 0x7F. Status bajty nabývají hodnot 0x80 až 0xFF (Ellinger, 2014). Endianita (pořadí bajtů) SMF je Big Endian. Big endian znamená, že při čtení čísla složeného z více bajtů první bajt je MSB (nejvíce významný bajt) a poslední bajt je LSB (nejméně významný bajt).

2.1.2 Header Chunk

Každý MIDI soubor musí začínat tzv. Header Chunkem, který se vždy dohromady skládá z 14 bajtů. První čtyři mají hexadecimální hodnotu 0x49546864, což je doslovný string "MThd". Následující čtyři bajty udávají zbývající délku header chunku, což je vždy hexadecimálních 0x00000006. Následující dva bajty udávají typ MIDI souboru. Existují tři typy MIDI souborů. Typ 0 s pouze jednou stopou, typ 1 s více stopami a typ 2 s více celými písněmi. Další 2 bajty specifikují kolik následuje stop. Tento údaj je přítomen i u souborů typu 0 a je v tom případě je vždy 1. Poslední dva bajty specifikují jednotku pro delta časování. Hodnota udává počet tiků za čtvrtnotu. Po konci Header chunku se okamžitě přechází do Track Chunku (Stanford).

2.1.3 Track Chunk

Každý Track chunk začíná čtyř-bajtovou hodnotou hexadecimálně 0x4d54726b, což je doslovný string "MTrk". Po této zprávě následuje specifikace délky stopy v dalších čtyřech bajtech, které vyjadřují unsigned long. Následují již stopové události, které vždy začínají delta časem jako počet tiků (Stanford).

Obrázek 1 Struktura Header a Track chunků

Header Chunk				
Type of Chunk (MThd in ASCII)	Length (bytes)	Format	Number of Track Chunks	Division (ticks)
4D 54 68 64	00 00 00 06	00 01	00 05	00 60

(a)

Track Chunk						
Type of Chunk (MTrk in ASCII)	Length (bytes)	Delta Time ₁ (ticks)	Event ₁	...	Delta Time _n (ticks)	Event _n
4D 54 72 6B	00 00 01 3B	00	C1 2E	...	83 00	FF 2F 00

(b)

Zdroj: (Yi-Hsin, a další, 2020)

Pokud je tato hodnota menší jak 0x7F, dekadické 127 je další bajt také bajtem delta času. Hodnoty se poté spojují tak, že se odebere MSB (nejvíce významný bit) obou bajtů a vytvoří se 14-ti bitové číslo např. přijde bajt 0x81 a poté 0x00. 0x81 je binárně 10000001. Odebere se MSB a vzniklé číslo se spojí s 0000000. Nakonec vzniká 14-ti bitové číslo 00000010000000, které se rovná dekadickému 128. Tomuto se říká Variable length variable (VLV). Po stanovení delta času mezi událostmi přichází status bajt určující, co se má stát za událost (Sapp). Události začínající na hodnotách 0x80 až 0xFE jsou obyčejné příkazy referující vždy na nějakou událost týkající se přímo přehrávání not (Ellinger, 2014).

Obrázek 2 Seznam obyčejných příkazů

Message	Status 80-FF	Number data bytes	Data 1 (0-127)	Data 2 (0-127)
Note Off NOF	8n	2	Note Number	Velocity (ignored)
Note On NON	9n	2	Note Number	Velocity
Aftertouch(Poly)	An	2	Note Number	Pressure Value
Control Change	Bn	2	Control Number	Control Value
Program Change	Cn	1	Patch Number	Not Used
Aftertouch (Channel)	Dn	1	Note Number	Pressure Value
Pitch Wheel	En	2	Pitch Bend LSB	Pitch Bend MSB
System Exclusive	Fx	varies	Mfr. specific	varies

n = channels 0-F
x = 0-F but is not related to channel

Zdroj: (Ellinger, 2014)

Příkazy začínající 0xFF jsou META události a po jejich přijetí následuje ještě jeden bajt specifikující jaká META událost se má stát (Stanford).

Obrázek 3 Seznam META událostí

Type	Event	Type	Event
0x00	Sequence number	0x20	MIDI channel prefix assignment
0x01	Text event	0x2F	End of track
0x02	Copyright notice	0x51	Tempo setting
0x03	Sequence or track name	0x54	SMPTE offset
0x04	Instrument name	0x58	Time signature
0x05	Lyric text	0x59	Key signature
0x06	Marker text	0x7F	Sequencer specific event
0x07	Cue point		

Zdroj: (Stanford)

Running status

Pokud po přijetí informace o delta časování nepřijde status bajt, ale data bajt předpokládá se, že se má stát událost stejná jako událost minulá, a proto si musí zařízení vždy pamatovat minulou událost (Neznámý).

2.2 Výběr řešení

Základem celého výrobku je xylofon o mnou vybrané minimální velikosti 32 kamenů. Nad každým kamenem je solenoid, který do něj udeřuje. Protože je 32 solenoidů příliš velký počet pro ovládání jednotlivými piny mikrokontroleru, zvolil jsem metodu, kde využívám čtyř sériově spojených osmibitových posuvných registrů, do kterých posílám jedno 32bitové číslo, kde každý bit koresponduje s jedním solenoidem. Pro spínání solenoidů využívám tranzistorových polí.

2.3 Výběr a popis součástek

Xylofon

Samotným xylofonem je zvonkohra o velikosti 34 (20 + 14) kamenů od firmy G+W Cheb. Kvůli technickým limitacím zmíněným později v této kapitole je využito pouze 32 kamenů.

Obrázek 4 Zvonkohra G+W

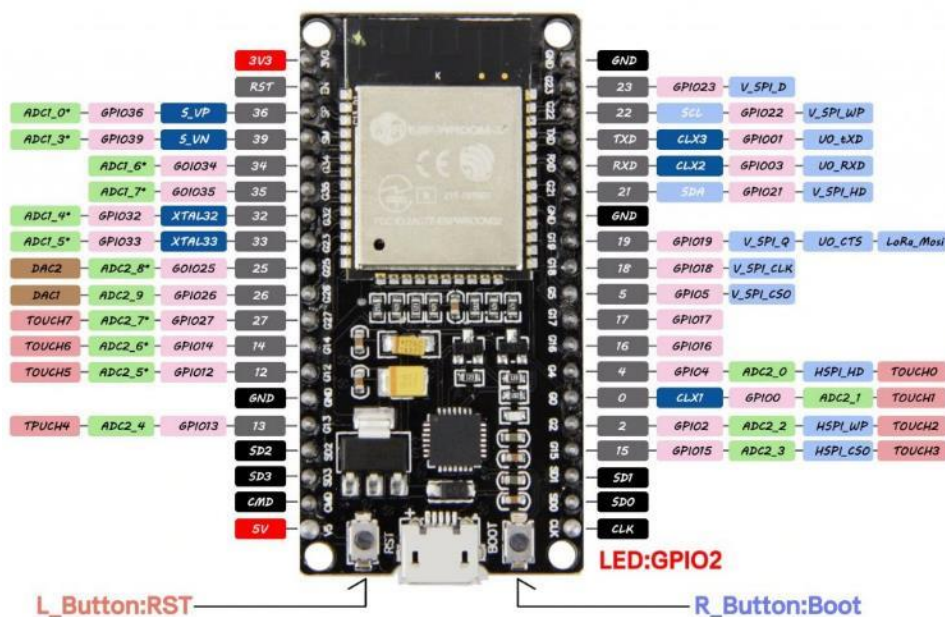


Zdroj: (Kytary.cz)

Mikrokontroler

Jako řídicí jednotku jsem zvolil mikrokontroler ESP32, protože disponuje schopností bezdrátové komunikace pomocí Bluetooth nebo Wi-Fi, vysokou taktovací frekvencí až 240 MHz a dvěma jádry. Disponuje taky velkým počtem GPIO (General purpose input output) pinů a to 34. **Mikrokontroler je napájen a pracuje s logickými úrovněmi 3.3 V.

Obrázek 5 Rozložení pinů modulu ESP32



Zdroj: (LaskaKit.cz)

Solenoidy

Solenoidy (česky označovány také jako tažné elektromagnety), jsou hustě navinuté cívky s pohyblivým feromagnetickým jádrem a pružinou, která jádro vrací do původní pozice po rozpojení obvodu. Solenoidy, které jsem vybral jsou JF-0530B, které jsou navrženy pro spínání na 12 V při odběru 300 mA. V konstrukci solenoidů jsou vyvrtány díry s metrickými závity M3, které usnadňují upevňování do profilů, které solenoidy drží.

Posuvné registry

Posuvné registry (Anglicky také Shift-bit register), jsou integrované obvody, které převádějí sériová data přicházející na vstup na paralelní výstup dat. Při náběžné hraně časovacího signálu na časovací vstup obvodu je stav na vstupu promítnut na první výstup. Při další náběžné hraně je opět vstup promítnut na první výstup, ale zároveň je původní stav prvního výstupu promítnut na druhý výstup. Tento posuv se opakuje do té doby, než bit dojde na konec registru a je zapomenut nebo předán na vstup dalšího registru zapojeného v sérii. Já jsem zvolil posuvné registry SN74HC595 od firmy Texas Instruments, které mají výstupní status registr a tedy posouvání bity se nebudou cestou na jejich místo okamžitě promítat na výstupy a interferovat s ostatními solenoidy. Tento čip je však pouze 8bitový a je nutné tedy spojit čtyři tyto čipy v sérii. Toto je umožněno, protože zvolený čip posuvného registru disponuje pinem s bitem přetečení, který se připojí na vstup následujícího registru (Texas Instrument, 2023).

Tranzistorová pole

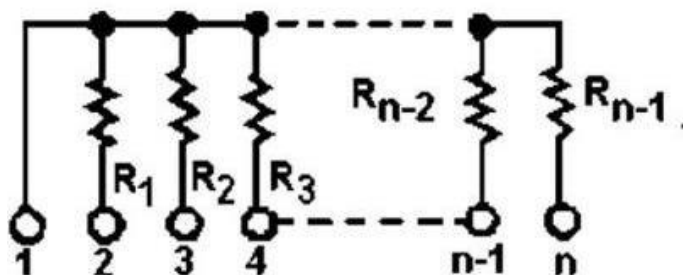
Tranzistorová pole jsou integrované obvody obsahující určitý počet výkonových tranzistorů. Každý tranzistor má oddělenou bázi a společný emitor nebo kolektor, podle výrobce. Jako tranzistorová pole jsem zvolil TD62083AP s osmi výkonovými bipolárními darlingtonovými tranzistory od firmy Toshiba. Darlingtonovy tranzistory spínají proud až 500 mA na jeden výstup, což postačí pro spínání zvolených solenoidů, které odebírají 300 mA. Tento čip také disponuje i vnitřními reverzními diodami, které jsou vhodné přidávat k indukčním zátěžím jako jsou například solenoidy. Kolektory jednotlivých tranzistorů jsou oddělené a Emitory jsou společné. Tranzistory spínají oproti zemi. Dle datasheetu TD62083AP stačí na plné otevření tranzistoru 1 mA a proto jsem tedy umístil mezi výstupy posuvného registru a vstupy tranzistorového pole sériový odpor, který omezuje proud na 1 mA (Toshiba, 2009). V tomto případě je výstupní napětí registrů rovno napětí napájecímu,

které je v mém případě 3.3 V, protože mnou zvolený mikrokontroler ESP32 pracuje s logickými hodnotami 3.3V (Texas Instrument, 2023). Odpor má tedy velikost 3.3kΩ.

$$R = \frac{U}{I} = \frac{3.3}{10^{-3}} = 3.3 \cdot 10^3 \Omega = 3.3 \text{ k}\Omega$$

Pro ušetření místa na desce plošného spoje volím tyto rezistory v SMD provedení. Aby bylo zajištěno, že v případě log. 0 na výstupu registru nezůstane báze tranzistorů plovoucí je na výstupy registrů ještě připojen tzv. Pull Down rezistor, který je připojen k zemi. Velikost odporu jsem zvolil 10kΩ a provedení rezistorů jsem zvolil jako rezistorovou síť typu RRA s osmi rezistory s jedním společným pinem.

Obrázek 6 Vnitřní schéma rezistorové sítě RRA

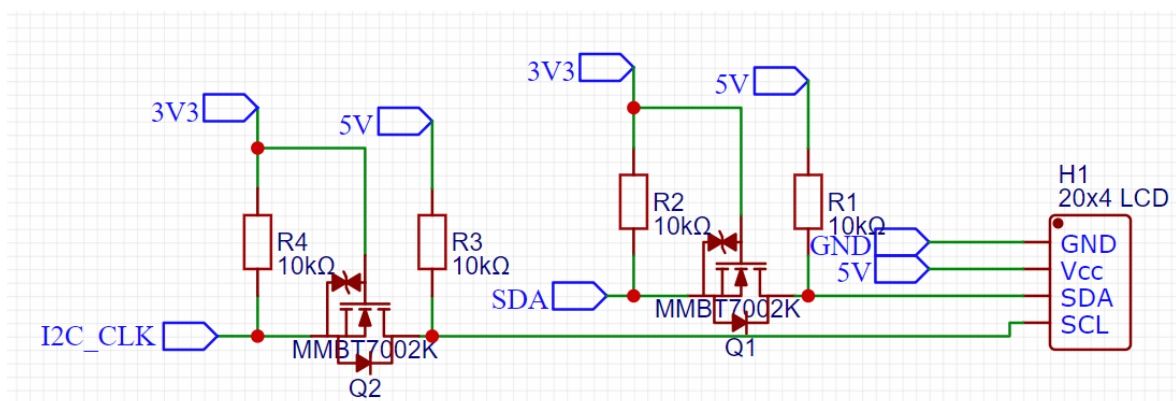


Zdroj: (GM Electronic)

Periferie

Jako úložiště jsem zvolil modul microSD karet z důvodu velké dostupnosti a rozšířenosti. Modul jsem volil takový, aby byl napájen stejným napětím jako mikrokontroler a to 3.3V. S modulem komunikuje mikrokontroler pomocí sběrnice SPI. Jako uživatelské rozhraní jsem zvolil charakterovou LCD obrazovku o velikosti 4x20 charakterů a rotační enkodér pro navigaci. LCD obrazovka je vybavena I²C převodníkem pro usnadnění komunikace s obrazovkou. Rotací enkodéru se uživatel posouvá skrze menu nahoru nebo dolů a stiskem svůj výběr potvrzuje. Protože LCD obrazovka je napájena napětím 5 V a pracuje tedy také s logickými úrovněmi 5 V. Jako unipolární tranzistor pro oboustranný měnič logických úrovní jsem zvolil MMBT7002K, kvůli jeho rychlým spínacím charakteristikám, kde čas sepnutí i vypnutí je typicky 20 ns (Diotec Semiconductor AG). Rezistory i tranzistory jsem zvolil malé SMD, protože není třeba aby byly výkonové a ušetřím tak místo na desce plošného spoje.

Obrázek 7 Schéma převaděče logických úrovní



Zdroj: Vlastní tvorba

2.4 Výpočet ceny projektu

Xylofon	1 200 Kč	Posuvné registry	32 Kč
DPS	520 Kč	Tranzistorová pole	84 Kč
Solenoidy	3 350 Kč	Drobná elektronika	600 Kč
Mikrokontroler	208 Kč	Spojovací materiál	50 Kč
LCD obrazovka	208 Kč	Napájecí zdroj	500 Kč
microSD karta	169 Kč	3D tiskový filament	100 Kč
Hliníkové profily	200 Kč		

Dohromady celý projekt stál přibližně 7 221 Kč.

3 Praktická část práce

V této kapitole se věnuji praktickým řešením problémů a implementací řešení z teoretické části práce.

3.1 Upevnění solenoidů

Solenoidy jsou upevněny na hliníkových L profilech o velikosti 40x20 mm, které jsou zasunuty a zašroubovány do 3D tisknutých násad, nasunutých z obou stran xylofonu. Na zadních stranách profilů jsou kabelové lišty, kterými jsou vedeny všechny dráty od solenoidů. Do profilu jsou vyvrtány díry po stranách pro upevnění do 3D tisknutých násad a také po celé délce pro připevnění solenoidů. Připevněny jsou dvěma metrickými šrouby M3x6, zkrácenými jednou maticí, protože moc závit je moc dlouhý ale kratší šrouby se shánějí složitě a je levnější sehnat šrouby delší a poté je takto zkrátit. Všechny části na xylofonu jsem designoval nedestruktivně pro samotné tělo xylofonu a po odšroubování profilů z držáků jdou sundat bez jakéhokoliv poškození způsobeného pro xylofon.

Obrázek 8 Upevnění solenoidů a vedení drátů



Zdroj: Vlastní tvorba

3.2 Deska plošného spoje

Schéma obvodu a desku plošného spoje jsem designoval v programu EasyEDA. Plošný spoj je oboustranný. Po objednání desky z plosnaky.cz jsem celý plošný spoj osadil a napájel. Komunikace s ovládacím modulem je zařízena konektory na zadní straně desky s nezaměnitelným směrem připojení, aby nemohlo dojít k přepólování periférií. Pro testování funkce tranzistorů bez připojených solenoidů a jsou jako indikace paralelně ke každému solenoidu připojeny LED diody. Návrh plošného spoje viz. Příloha 1. Schéma zapojení viz. Příloha 2.

3.3 Ovládací modul

Základem modelu je model LCD-cover-ORIGINAL-MK3 od Prusa Research, který jsem upravil pro velikosti svých součástí a mé potřeby (Prusa Research, 2021). Na zadní straně modelu je trojúhelníková lišta, která se zasouvá do druhé části kolejnice připevněné ke xylofonu oboustrannou lepící páskou. Modul je tedy odepínatelný a dá se na něm lehce pracovat. Modul je vybaven charakterovou LCD obrazovkou o velikosti 20x4 charakterů s I2C převodníkem pro usnadnění komunikace, modulem pro čtení microSD karet a rotačním enkodérem pro navigaci skrze menu. Stejně jako při připevňování solenoidu ovládací modul nepoškozuje samotné tělo xylofonu.

3.4 Řídící program

Program je psán v Arduino Framework pomocí vývojového prostředí PlatformIO ve Visual Studio Code.

3.4.1 Vypisování na obrazovku

Komunikace s obrazovkou je zajištěna pomocí LiquidCrystalI2C knihovny. Obrazovka se aktualizuje pouze pokud nastane nějaká změna, aby obrazovka neblíkala. Uživatelské rozhraní je rozděleno do většího počtu sub menu, skrze které uživatel naviguje pomocí rotačního enkodéru. Aktuální pozice kurzoru je vždy znázorněna symbolem '>'. V mnoha submenu jsou implementovány upozornění např. když se uživatel snaží přijmout soubor bezdrátově bez zapnutého Bluetooth rozhraní je na tento fakt upozorněn.

3.4.2 Čtení dat z microSD karty

Protože microSD karta může obsahovat nekonečně mnoho souborů omezena pouze velikostí SD karty musí být funkce škálovatelná. Funkce vždy vypíše na obrazovku celou složku počínaje v kořenovém adresáři. Stiskem na rotačním enkodéru se potvrdí výběr souboru, pokud je však soubor vyšší složka je funkce zavolána znovu o složku výše, což lze neomezeně krát rekurzivně opakovat. Pokud si uživatel vybere soubor je uložena jeho cesta k umístění souboru, která je poté předávána funkci pro parsování MIDI souborů.

3.4.3 Implementace parsování

Po výběru souboru pro přehrání je nejdříve zkontrolováno, zda je soubor v MIDI formátu. Pokud není, je parsování ukončeno a uživatel upozorněn. Poté jsou přečteny všechny základní informace o typu a velikosti souboru. Pokud jde o formát MIDI typu 2, je parsování ukončeno, protože není možné hrát více písní najednou. Po kontrole těchto parametrů funkce

prochází soubor bajt po bajtu a ukládá si delta časování + událost, která má v tom momentě nastat. I když nějaké události nemá smysl, nebo není ani možné využít, musí je mikrokontroler stejně přečíst, protože MIDI file formát je tvořen tak aby byl čten sériově a mohlo by dojít k problémům s časováním. Informace o času se ukládá pouze ve formátu PPQ (Pulses per Quaternote) a delta časování v milisekundách je vypočítáno až přímo při přehrávání.

3.4.4 Přehrávání

Po výběru souboru a stisknutí volby pro přehrávání je provedeno parsování souboru a po jeho ukončení se začne přehrávat. Posílání informace o sepnutí solenoidů se posílá pomocí mnou upravené zabudované arduino funkce shiftOut. Původní shiftOut přijímá pouze jeden bajt, ale já potřebuji posílat 32bitová čísla (4 bajty) a také na konci posílání má funkce sepnout výstupní status registr a promítnout ho na výstup. Po přečtení všech událostí je přehrávání ukončeno.

3.4.5 Bluetooth

Pro umožnění bezdrátové Bluetooth komunikace je využita knihovna BluetoothSerial. Nejdříve musí uživatel v menu Bluetooth zapnout a vybrat jím požadovaný mód. Pokud vybere příjem souborů ESP32 začne poslouchat na Bluetooth sériové lince a po ukončení přenosu, je uživatel upozorněn na obrazovce, že přenos byl ukončen. Jelikož je výchozí nastavení přijímacího bufferu v knihovně BluetoothSerial nedostačujících 512 B a přenášená data se při přenosu větších souborů ztrácejí, změnil jsem ho na 8 KB, aby se do něj vešly i MIDI soubory, které se většinou pohybují okolo 3KB pro klasické písně. ESP32 disponuje 16 MB paměti, takže dopad na rychlost zpracovávání a práci se zbytkem kódu by měl být zanedbatelný. Pokud uživatel zvolí možnost přímého hraní začne xylofon poslouchat na sériové lince a podle přicházejících dat spíná jednotlivé solenoidy.

3.5 Mobilní aplikace

Programováno v Android studio. Základem programu je projekt Kotlin-Bluetooth od uživatele appsinthesky, ale tento program musel být upraven, kvůli novější verzi Android Studia a zpětné nekompatibilitě (Appsinthesky, 2018). Po zapnutí aplikace je nejprve uživateli nabídnut výběr ze všech spárovaných zařízení s jeho mobilním zařízením. Po výběru připojení na ESP32 má na výběr poslat soubor, přímo ovládat xylofon anebo se odpojit. Pokud chce uživatel poslat soubor, otevře se Android file_open_dialog, což je

zabudované okno výběru souboru v systému Android a po výběru souboru je soubor odeslán do ESP32. Pokud je vybrána možnost přímého ovládání je uživateli prezentována obrazovka s 32 tlačítky, každé reprezentující jeden kamen xylofonu.

Obrázek 9 Obrazovka výběru akce



Zdroj: Vlastní tvorba

Obrázek 10 Obrazovka přímého hraní



Zdroj: Vlastní tvorba

3.6 Napájení výrobku

Celý výrobek je napájen 12 V zásuvkovým spínaným zdrojem. Do výrobku se zdroj připojuje pomocí 5.5x2.1 mm DC barell jacku. Těmito 12 V jsou napájeny solenoidy, které

všechny mají společné plus a jsou spínány proti zemi. Na desce plošného spoje je tzv. step down měnič, spínaný zdroj, který snižuje napětí 12 V na 5 V pro napájení LCD obrazovky a mikrořadiče ESP32. Modul mikrořadiče ESP32 je vybaven vlastním stabilizátorem na 3.3 V. Tímto stabilizátorem jsou poté napájeny posuvné registry, rotační enkodér a modul microSD karty. Protože při zapojení výrobku a připojení napájení na posuvné registry se všechny výstupy automaticky sepnou a tím pádem se solenoidy stlačí, čímž zatíží zdroj na tolik že se vypne, přidal jsem na zadní stranu přepínač, který přidržetím odpojí solenoidy od napájení a tím pádem dovolí mikrokontroleru se inicializovat a výstupy registrů vypnout.

3.7 3D tisk

Všechny modely jsou modelovány nebo upraveny mnou v programu Autodesk Fusion 360. Modely jsem poté vytiskl na mé 3D tiskárně Prusa Mini+ z PLA materiálu. Tento materiál je levný a pro mé potřeby má vhodné parametry.

4 Závěr

Cílem práce bylo sestavit výrobek schopný přehrávat soubory formátu MIDI na Xylofonu. Výrobek byl designován tak, aby šel z xylofonu sundat a xylofon zůstal naprosto nepoškozen. Dále bylo nutné vytvořit aplikaci, která bude do výrobku schopna posílat informace, nebo xylofon přímo ovládat.

Nejprve bylo potřeba sestavit samotnou konstrukci a připevnit solenoidy na xylofon. V této části se mnoho obtíží neobjevilo. Největším úskalím bylo vymyslet vedení drátů, což bylo nakonec vyřešeno pomocí elektrikářských lišt.

Dalším úkolem bylo vytvořit design plošného spoje a elektronické zapojení všech součástek. Plošný spoj jsem úspěšně na designoval a nechal si vyrobit. Plošný spoj je však oboustranný a z výroby nebyl prokoven a tudíž strany spoje nebyly spojeny. Prokovení jsem samostatně dodělával a tím jsem strany propojil.

Dalším krokem bylo naprogramování řídicí jednotky, Mikrokontroleru ESP32. Tato část obnášela mnoho překážek. Největším problémem samozřejmě bylo samotné čtení a přehrávání MIDI souborů. V této části byl problém hlavně s endianitou MIDI, která je opačná oproti ESP32, takže pořadí bajtů musí být otáčeno samostatně. Také jsem musel vyřešit, jak být připraven vypsát teoreticky nekonečně velkou složku na charakterovou obrazovku o čtyřech řádcích a dvaceti sloupcích. I přes všechny tyto zádrhly jsem nakonec vše zprovoznil a řídicí program je funkční a dělá to, co podle zadání má.

Posledním cílem zadání bylo vytvořit mobilní aplikaci pro komunikaci a ovládání výrobku. Hlavní překážkou v této části byla práce s vývojovým prostředím Android Studio a programovacím jazykem Kotlin, protože oba tyto prvky se stále vyvíjejí a často se mění. Bylo proto složité hledat dokumentace na internetu, protože mnoho existujících příkladů je nefunkčních. Cíle však bylo nakonec dosaženo.

Výrobek je tedy funkční a měl by splňovat všechny body zadání. Jsem rád, že jsem si vybral jako možnost dělat dlouhodobou maturitní práci, protože cítím, že mi přinesla mnoho nových praktických i teoretických znalostí v mém oboru.

Nakonec bych ještě chtěl poděkovat vedoucímu práce panu Ing. Davidu Kubátovi, hlavně s pomocí při tvorbě mobilní aplikace.

5 Seznam použitých zdrojů

- Appsinthesky. 2018.** GitHub - appsinthesky/Kotlin-Bluetooth. [Online] 16. Leden 2018. [Citace: 24. Listopad 2022.] <https://github.com/appsinthesky/Kotlin-Bluetooth>.
- Diotec Semiconductor AG.** mmbt7002k-2577485.pdf. [Online] [Citace: 12. Únor 2023.] <https://cz.mouser.com/datasheet/2/849/mmbt7002k-2577485.pdf>.
- Ellinger, John. 2014.** MIDI Basics. [Online] 2014. [Citace: 10. Únor 2023.] https://people.carleton.edu/~jellinge/m208w14/pdf/02MIDIBasics_doc.pdf.
- GM Electronic.** GM electronic | elektronické součástky, komponenty - RRA 8X10k 2% rezistorová síť. [Online] [Citace: 20. Únor 2023.] <https://www.gme.cz/v/1494455/rra-8x10k-2-rezistorova-sit>.
- Kytary.cz.** GW ZA 20+14 Zvonkohra. [Online] [Citace: 20. Únor 2023.] <https://kytary.cz/gw-za-20-14/HN144590/>.
- LaskaKit.cz.** IoT ESP-WROOM-32 2.4GHz Dual-Mode WiFi+Bluetooth rev.1, CP2102 | LaskaKit. [Online] [Citace: 10. Únor 2023.] <https://www.laskakit.cz/iot-esp-32s-2-4ghz-dual-mode-wifi-bluetooth-rev-1--cp2102/>.
- Neznámý.** MIDI Specification: Running Status. [Online] [Citace: 10. Únor 2023.] <http://midi.teragonaudio.com/tech/midispec/run.htm>.
- Prusa Research. 2021.** i3 MK3 Printable parts od autora Prusa Research | Stáhněte si zdarma STL model | Printables.com. [Online] 22. Listopad 2021. [Citace: 14. Listopad 2022.] <https://www.printables.com/cs/model/88240-i3-mk3-printable-parts/files>.
- Sapp, Craig Stuart.** Variable Length Values. [Online] [Citace: 10. Únor 2023.] <http://www.ccarh.org/courses/253/handout/vlv/>.
- Stanford.** Standard MIDI file structure. [Online] [Citace: 10. Únor 2023.] <https://ccrma.stanford.edu/~craig/14q/midifile/MidiFileFormat.html>.
- Texas Instrument. 2023.** SNx4HC595 8-Bit Shift Registers With 3-State Output Registers datasheet (Rev. J). [Online] 2023. [Citace: 12. Únor 2023.] <https://www.ti.com/lit/ds/symlink/sn74hc595.pdf>.
- Toshiba. 2009.** TD62083~084APG/AFG. [Online] 1. Říjen 2009. [Citace: 12. Únor 2023.] https://cz.mouser.com/datasheet/2/408/TD62083AFG_datasheet_en_20091001-875843.pdf.
- Yi-Hsin, Liu a Da-Chun, Wu. 2020.** The chunk structures of a MIDI file and instances with hexadecimal... | Download Scientific Diagram. [Online] Červenec 2020. [Citace: 16. Únor 2023.] https://www.researchgate.net/figure/The-chunk-structures-of-a-MIDI-file-and-instances-with-hexadecimal-representations-a_fig4_338777386.

6 Seznam příloh

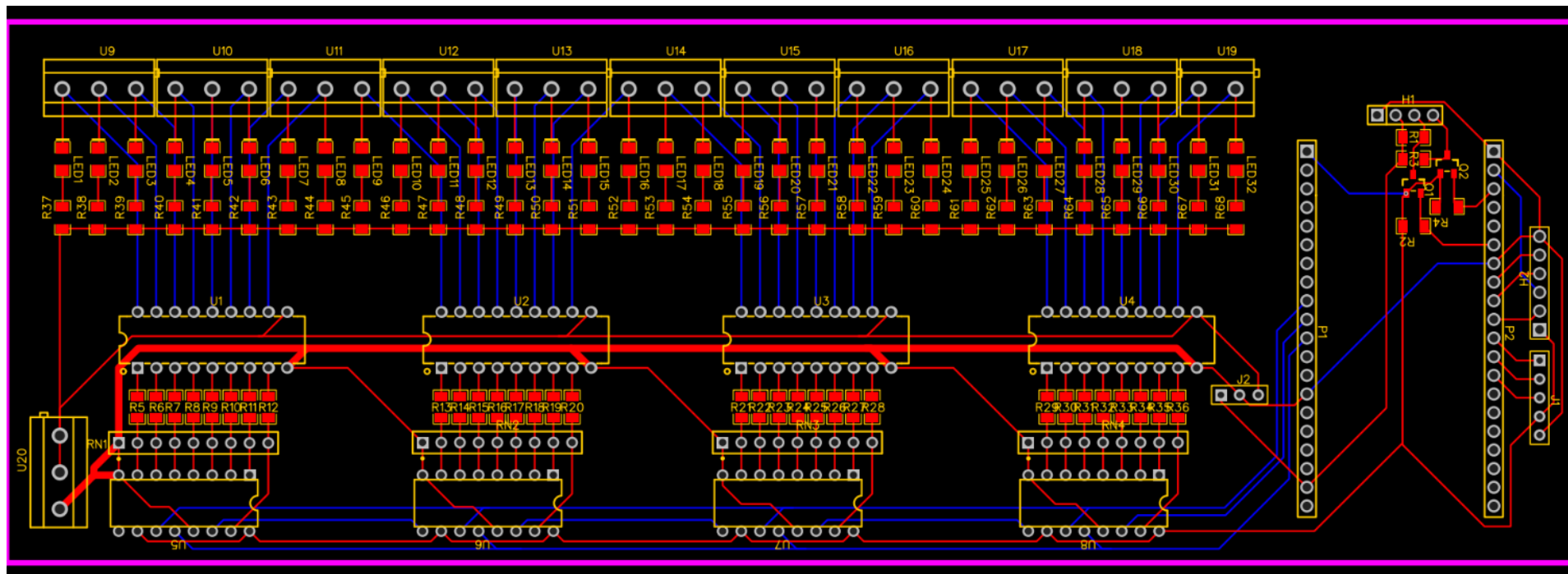
Příloha č. 1: Návrh plošného spoje

Příloha č. 2: Schéma zapojení plošného spoje

Seznam obrázků

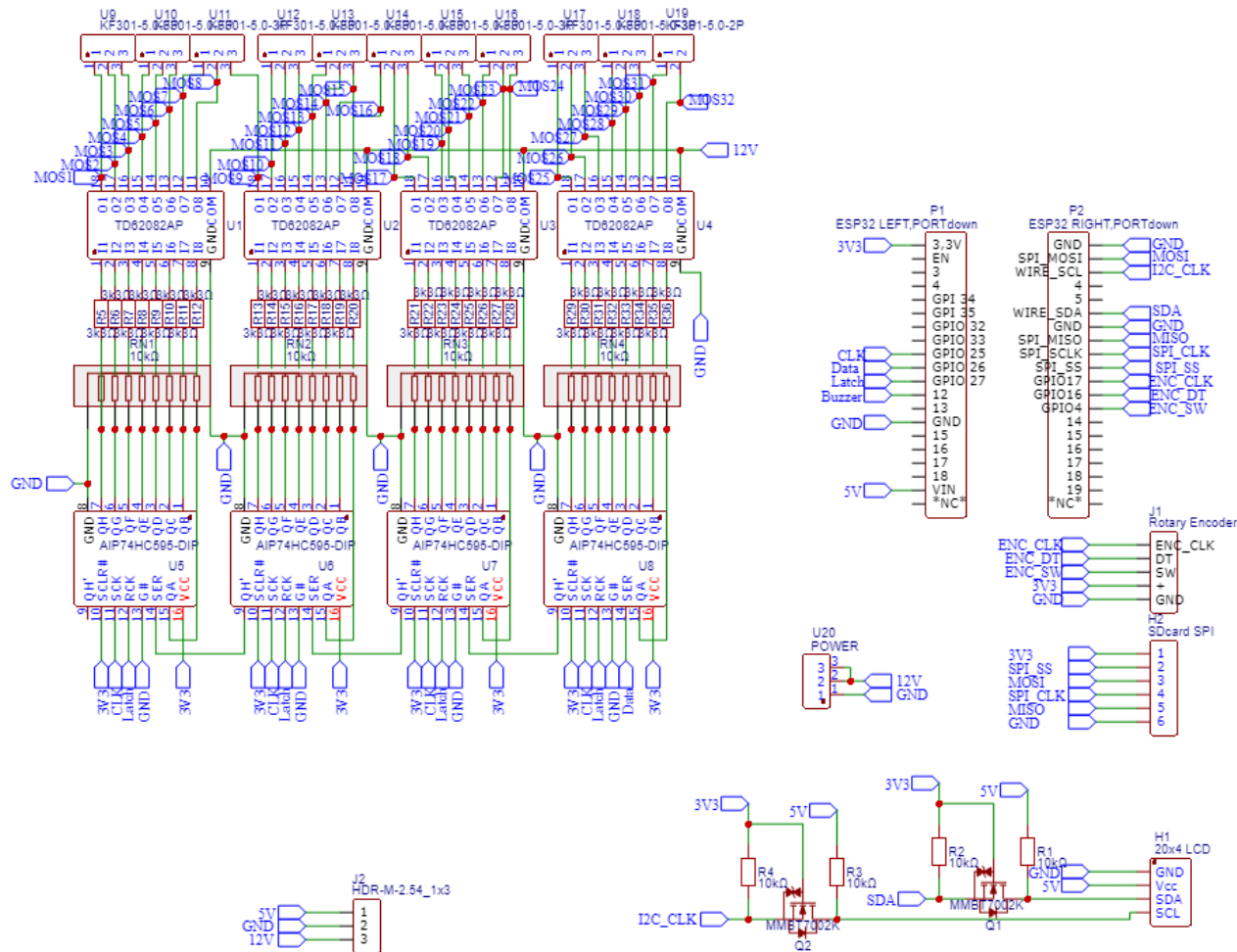
Obrázek 1	Struktura Header a Track chunků	7
Obrázek 2	Seznam obyčejných příkazů.....	7
Obrázek 3	Seznam META událostí	8
Obrázek 4	Zvonkohra G+W	9
Obrázek 5	Rozložení pinů modulu ESP32	9
Obrázek 6	Vnitřní schéma rezistorové sítě RRA.....	11
Obrázek 7	Schéma převaděče logických úrovní.....	12
Obrázek 8	Upevnění solenoidů a vedení drátů.....	13
Obrázek 9	Obrazovka výběru akce.....	16
Obrázek 10	Obrazovka přímého hraní.....	16

Příloha č.1:



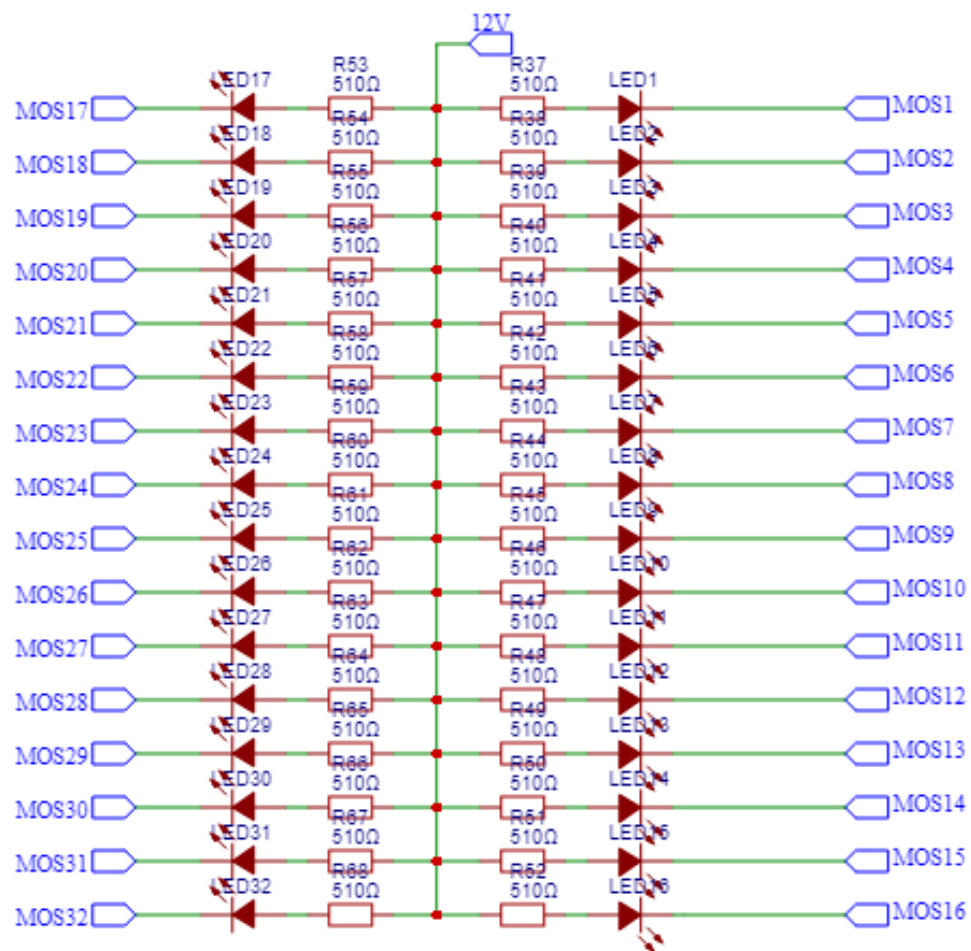
Zdroj: Vlastní tvorba

Příloha č.2:



Zdroj: Vlastní tvorba

Příloha č.2:



Zdroj: Vlastní tvorba