



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

**VESTAVĚNÉ ZAŘÍZENÍ PRO OVLÁDÁNÍ DIGITÁLNÍ
AUDIO STANICE**

EMBEDDED DEVICE FOR CONTROL OF DIGITAL AUDIO WORKSTATION

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. TOMÁŠ SVOBODA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. VÁCLAV ŠIMEK

BRNO 2019

Zadání diplomové práce



22105

Student: **Svoboda Tomáš, Bc.**
Program: Informační technologie Obor: Počítačové a vestavěné systémy
Název: **Vestavěné zařízení pro ovládání digitální audio stanice**
Embedded Device for Control of Digital Audio Workstation
Kategorie: Vestavěné systémy

Zadání:

1. Seznamte se s existujícími řešeními pro ovládání DAW (Digital Audio Workstation) softwaru prostřednictvím fyzického kontroléru.
2. Prostudujte protokol pro přenos dat mezi kontrolérem a DAW prostředím. Dále nastudujte problematiku řízení DC motoru motorizovaného potenciometru.
3. Navrhněte architekturu vestavěného zařízení pro ovládání DAW softwaru. Zvolte vhodné HW komponenty pro jeho realizaci.
4. Připravte detailní blokové schéma jednotlivých funkčních celků navrženého zařízení.
5. V návrhovém editoru desek plošných spojů připravte obvodovou realizaci těchto bloků, fyzicky je implementujte a následně proved'te jejich základní oživení.
6. Implementujte firmware nezbytný k zajištění požadované funkčnosti jednotlivých částí navrženého systému.
7. Vhodným způsobem demonstруйте funkčnost navrženého řešení a diskutujte možnost dalšího rozšíření.
8. Zhodnoťte dosažené výsledky.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 1 až 4 zadání.

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Šimek Václav, Ing.**
Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.
Datum zadání: 1. listopadu 2018
Datum odevzdání: 22. května 2019
Datum schválení: 1. května 2019

Abstrakt

Cílem této práce je navrhnout architekturu vestavěného zařízení, které bude použito v nahrávacím studiu pro ovládání DAW softwaru. Práce nejprve shrnuje potřebné znalosti k návrhu tohoto druhu zařízení, včetně krátké rešerše existujících řešení a popisu protokolů, které lze použít pro komunikaci s nahrávacím softwarem. Následně je v práci navržena architektura zařízení, která problém dekomponuje do několika modulů. V rámci práce jsou navrženy a vyrobeny dva hardwarové moduly, z nichž každý je tvořen samostatnou deskou plošných spojů s vlastním mikrokontrolérem. Pro každý z mikrokontrolérů byl v rámci práce vytvořen ovládací firmware. V závěru se práce zabývá výrobou hliníkového panelu, do kterého jsou umístěny vytvořené moduly. Výsledkem práce je prototyp zařízení, které může být použito k ovládání DAW softwaru.

Abstract

The aim of this work is to design an architecture of the embedded device that will be used for controlling DAW software in recording studio. First of all, attention is given to a brief summary of the necessary knowledge which is needed to design such kind of device. After that follows short survey of the existing solutions and description of protocols which can be used for communication with the recording software. Then, subsequent part of the thesis builds upon these foundations and further elaborates the device architecture by means of decomposing it into several modules. In fact, two hardware modules are designed and manufactured, when each of them is conceived on a separate PCB with its own microcontroller. Then the control firmware has been implemented for each of the modules. At the end of the work an aluminium enclosure, which holds both modules, is designed. The result of this work is a functional prototype of the assembled controller which can be used for the purpose of controlling DAW software.

Klíčová slova

DAW, kontrolér, MIDI, ovládací zařízení, digitální audio stanice, nahrávací software, motorizovaný potenciometr

Keywords

DAW, controller, MIDI, control device, digital audio workstation, recording software, motorized potentiometer

Citace

SVOBODA, Tomáš. *Vestavěné zařízení pro ovládání digitální audio stanice*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Václav Šimek

Vestavěné zařízení pro ovládání digitální audio stanice

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením pana Ing. Václava Šimka. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Tomáš Svoboda

22. května 2019

Poděkování

Rád bych poděkoval vedoucímu práce, panu Ing. Václavu Šimkovi, za odbornou pomoc při návrhu, výrobě a osazení hardwarových modulů, 3D tisku potřebných součástí a za užitečné rady a připomínky při tvorbě této práce. Dále pak za zapůjčení potřebných přípravků pro zdárné dokončení práce. Velký dík patří i kolegům z práce za cenné rady a odbornou pomoc při návrhu desek plošných spojů, řešení problémů s jejich oživením a výrobě hliníkového panelu. V neposlední řadě bych rád poděkoval rodině, přítelkyni a kamarádům za podporu při tvorbě díla a za pomoc při jeho závěrečné korektuře.

Obsah

1	Úvod	3
2	Přehled existujících řešení	4
2.1	Termín digitální audio stanice	4
2.2	Zasazení kontroléru do kontextu nahrávání	5
2.3	Případové studie vybraných kontrolérů na trhu	6
2.4	Zhodnocení	9
3	Přenos dat mezi kontrolérem a PC	11
3.1	MIDI	11
3.2	Komunikační protokol	14
3.3	Spojení se stanicí	17
4	Návrh architektury zařízení	20
4.1	Motorizovaný potenciometr	20
4.2	Rozložení ovládacích prvků	23
4.3	Rozdělení do modulů	24
4.4	Zhodnocení	29
5	Návrh hardwarové platformy	30
5.1	Fader module	30
5.2	Line module	33
5.3	Testovací deska	35
6	Implementace firmware	36
6.1	Návrh architektury firmwaru	36
6.2	Komunikace	40
6.3	Přijímání MIDI zpráv	43
6.4	Ovladač AD převodníku	44
6.5	Ovladač displeje	46
7	Sestavení a testování zařízení	49
7.1	Výroba panelu zařízení	49
7.2	Testování zařízení	50
8	Závěr	52
	Literatura	53

A Schéma Fader modulu	55
B Schéma Line modulu	63
C DPS vytvořených modulů	70
D Nákres panelu zařízení	73
E Vyrobené zařízení	74

Kapitola 1

Úvod

Digitalizace přinesla do oblasti studiového nahrávání revoluci. Do té doby probíhalo nahrávání výhradně analogově. Od 70. let 20. století se digitální technologie postupně rozšiřovaly od velkých studií až po malé, kterým umožnily lepší podmínky pro rozvoj. Centrum dění se přesunulo z mnoha analogových zařízení do výkonného počítače, na němž jsou jednotlivé stopy nahrané a který dovoluje nad nimi simulovat všechna zařízení známá z analogového nahrávání, navíc přidává nové efekty a možnosti, jak pracovat se stopami, které nebyly v analogovém světě proveditelné. To vše se děje v softwaru – programu, který běží na počítači a který lze ovládat klávesnicí a myší, což není pro dlouhodobější práci ideální.

Cílem této práce je navrhnout a vytvořit DAW kontrolér pro ovládání nahrávacího softwaru. Kontrolér je vestavěné zařízení, které má audio inženýrům zjednodušit práci s nahrávacím programem a přiblížit pracovní postupy k těm, které byly používány na analogových mixážních pultech před digitalizací. Uživatelské rozhraní kontroléru má společné prvky s analogovými pulty, navíc však dovoluje využívat výhody, které digitální nahrávání nabízí.

Ve druhé kapitole bude nejprve představen pojem digitální audio stanice a kontroléru, který bude následně zasazen do kontextu nahrávání. Poté budou následovat krátké případové studie vybraných kontrolérů se zhodnocením klíčových vlastností. V kapitole 3 bude rozebrán přenos dat mezi kontrolérem a počítačem s důrazem na použité protokoly. Ve čtvrté kapitole bude představen motorizovaný potenciometr, který je hlavním prvkem uživatelského rozhraní zařízení. Následně bude podrobně popsán návrh architektury kontroléru. Pátá kapitola se bude zabývat návrhem, výrobou a oživením dvou hardwarových modulů, které jsou tvořeny dvěma deskami plošných spojů. V šesté kapitole bude rozebrána architektura vytvořeného firmwaru a detailně přiblíženy náročnější části implementace. Sedmá kapitola se bude zabývat výrobou hliníkového panelu, do kterého budou vytvořené moduly zasazeny.

Kapitola 2

Přehled existujících řešení

V této kapitole bude rozebrán termín digitální audio stanice a do celkového kontextu nahrávacího studia zasazen význam navrhovaného kontroléru. Dále zde bude představeno několik komerčně úspěšných řešení DAW kontrolérů na trhu.

2.1 Termín digitální audio stanice

Termín digitální audio stanice použitý v názvu práce vychází z anglického Digital Audio Workstation (dále jen DAW). Jedná se o nástroj, s jehož pomocí lze zaznamenávat a upravovat zvukové nahrávky. Jak již z názvu vyplývá, jedná se v tomto případě o digitální zpracování zvuku. Náplň této kapitoly čerpá především z [17].

V éře předcházející doslova všudypřítomnému rozšíření digitálních technologií byla nahrávací studia kompletně analogová. Zvuk ze zdroje (většinou mikrofonu) putoval skrz analogový mixážní pult, procesory, efekty do nahrávacího zařízení, kterým byl většinou vícestopý páskový magnetofon. V profesionálních studiích se jednalo především o magnetofony Studer, které používaly 2" pásku s 16 nebo 24 stopami. I za těchto podmínek však byly možnosti následného stříhu či komplexního zpracování velmi omezené. Výsledný zvuk se po nahrání všech stop míchal do volných stop na pásku nebo do jiného magnetofonu, odkud šel k dalšímu zpracování. Výsledná nahrávka byla přenesena na gramofonovou desku nebo kazetu.

Výhody digitálního zpracování jsou již na první pohled zřejmé. K dispozici je velké množství stop, do kterých lze nahrát jednotlivé zvuky, téměř neomezené možnosti stříhu, dotáčení oprav, automatizaci zeslabovačů a veškerých parametrů. Navíc je možno využít velké množství procesorů a efektů, které lze pro každou stopu libovolně kombinovat až do té míry, kterou zvládne stroj spočítat. Digitální audio stanice v sobě integruje nahrávací zařízení, mixážní pult, procesory, efekty, a to vše je následně k dispozici skrze ucelené v uživatelském rozhraní. Základem každého digitálního zpracování je kvalitní rozhraní z/do analogového světa – AD/DA převodník. K dalším významným aspektům bezesporu náleží vysoký výkon pracovní stanice, která zpracovává výsledný zvuk.

V zásadě existují dva typické přístupy k realizaci DAW: hardwarový a softwarový. Hardwarová varianta je řešení „vše v jednom“. Jedná se o zařízení, které v sobě kombinuje AD/DA převodníky, digitální zpracování zvuku uvnitř a jednodušší uživatelské rozhraní. Nevýhodou této platformy je její omezená rozšiřitelnost a většinou fixní počet stop, do kterých lze zvuky nahrát. Tato skutečnost je povětšinou dána omezenou výpočetní kapacitou počítače uvnitř. Omezené jsou také možnosti zpracování zvuku procesory a efekty, jelikož lze použít

jen vestavěné nástroje. Tento typ zařízení se používá převážně v malých nebo mobilních studiích, kde snadná přenositelnost představuje jednu z hlavních výhod.

Softwarové řešení naopak skýtá oproti hardwarové realizaci vyšší míru flexibility, na druhou stranu je však i po finanční stránce daleko nákladnější. Soustředí se totiž pouze na virtuální prostředí na obrazovce a na výpočet zvuku. Po hardwarové stránce je důležité mít výkonnou pracovní stanici, na které bude DAW software běžet. Velmi důležitá je také správná volba kvalitních AD/DA převodníků. V softwarovém řešení máme prakticky neomezený počet stop, které můžeme nahrát. Dále lze použít téměř libovolné množství zvukových procesorů a efektů. Vše je omezeno pouze výkonem pracovní stanice a použitým softwarem. Nutno podotknout, že veškeré zpracování zvuku probíhá číslicově uvnitř softwaru. Ten nabízí několik rozhraní pro rozšíření. Například existují standardní rozhraní pro vložení rozšíření třetí strany – zvukových efektů a procesorů.

Je tedy patrné, že příchod digitálních technologií se výrazně promítl do zpřístupnění pokročilých možností nahrávání i pro menší (tj. zejména domácí) studia. K vytvoření nahrávky prakticky stačí pouze výkonnější počítač, kvalitní AD/DA převodník (někdy označováno jako zvuková karta) a DAW software. Samozřejmostí je pak vybavení pro nahrávání a poslech. V této práci se budeme soustředit na použití softwarového DAW a na segment trhu pro malá nahrávací studia.

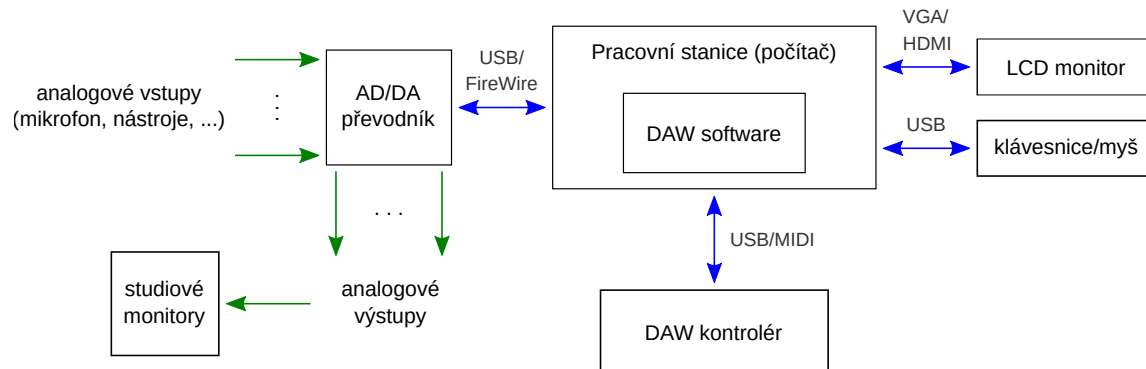
2.2 Zasazení kontroléru do kontextu nahrávání

Tato podkapitola čerpá z [14]. S příchodem digitálního nahrávání došlo k přenesení funkcí většiny přístrojů známých z analogového studia do oblasti softwarové realizace. Díky tomu máme k dispozici nástroje pro práci se zvukem na úrovni analogového studia, ovšem rozhraní, kterým se tyto nástroje ovládají, je diametrálně odlišné. Všechny informace jsou zobrazeny na monitoru a ovládání probíhá prostřednictvím klávesnice a myši. Pro domácí studia je tento způsob dostačující, ovšem pro každodenní práci v malém a středním studiu nejde o vyhovující způsob, a to hlavně z hlediska rychlosti mixování. A právě zde se nabízí prostor k uplatnění DAW kontroléru – tedy vestavěného zařízení, jehož návrh a fyzická realizace je náplní této práce. Díky němu může audio inženýr používat rozhraní podobné tomu z analogového studia společně s vlastnostmi digitální stanice. Vytvářený kontrolér se svoji velikostí a dostupnými funkcemi směřuje do malých nahrávacích studií. Ve velkých studiích se nachází mnohem větší kontroléry – ať už po stránce velikosti, tak i počtu nabízených funkcí.

V malých studiích je obvykle velmi omezený rozpočet. Z toho důvodu se vyplatí realizovat vše, kde je to technicky možné, na softwarovém základě. Základem je tedy DAW software společně s AD/DA převodníkem a poslechovými reproduktory – studiovými monitory. Dále je potřeba několik kvalitních mikrofonů a jiné vybavení studia. Hlavní roli, co se týče ovládání, zde hraje právě DAW kontrolér. Díky němu lze pohodlně ovládat software takřka bez sáhnutí na myš a využívat rozhraní známé z analogového světa.

Základní ovládací prvky kontroléru jsou společné s těmi, které lze nalézt na analogových pultech. Jedná se především o tahové potenciometry ovládající hlasitosti jednotlivých stop, dále pak o tlačítka pro ztišení/samostatné přehrávání stopy. Několik otočných potenciometrů z analogového pultu bylo sjednoceno do jednoho rotačního enkodéru, který se tváří jako nekonečný potenciometr (lze jím otáčet dokola), přičemž výběr editované hodnoty se provádí pomocí jiného enkodéru. V ovládací sekci jsou pak společná tlačítka pro ovládání přehrávání a pohybu v nahrávce, nově přibyla tlačítka pro ovládání automatizace digitálního nahrávání. Kontrolér je zařízení s konečným počtem zeslabovačů (tahových potenciometrů), které

ovšem může ovládat všechny virtuální stopy, které jsou v DAW softwaru. K tomu slouží speciální tlačítka na kontroléru, která umožňují posun okna ovládaných virtuálních stop. V každé stopě, kterou kontrolér obsahuje, je displej, na němž je zobrazen název ovládané stopy a aktuální hodnota nastavená nekonečným potenciometrem.



Obrázek 2.1: Zjednodušené schéma zapojení malého digitálního nahrávacího studia. Zelené šipky značí analogové hudební signály, modré digitální rozhraní. Z obrázku je patrné umístění DAW kontroléru i jeho způsob zapojení. Veškeré analogové hudební signály jsou zpracovány v AD/DA převodníku, který bývá typicky vícekanálový – jak vstup, tak i výstup. Převodník je spojen s pracovní stanicí, na které běží DAW software zpracovávající zvuk. Ten lze ovládat klávesnicí, myší nebo kontrolérem, který je k počítači připojen typicky přes MIDI, resp. USB/MIDI rozhraní.

2.3 Případové studie vybraných kontrolérů na trhu

V této podkapitole budou uvedeny 3 krátké případové studie vybraných DAW kontrolérů. Na trhu samozřejmě existuje široká škála takovýchto řešení. První vybraný kontrolér je špičkou ve své kategorii, druhý je komplexní zařízení, které do sebe integruje mnoho funkcí. Třetí vybrané zařízení je naopak jedním z nejlevnějších na trhu.

Mackie Control Unit Pro

V textu je čerpáno z informací a materiálů uvedených na stránkách výrobce [7]. Tento kontrolér byl vybrán, jelikož se jedná o špičkové zařízení v kategorii větších kontrolérů. Jedná se o pokročilejší verzi kontroléru, který pravděpodobně vychází ze staršího zařízení Mackie HUI. Kontrolér nabízí 8 stop k ovládnání. Každá stopa se skládá z motorizovaného potenciometru, tlačítek REC, SOLO, MUTE, SELECT a otočného rotačního enkodéru s prstencem led diod kolem pro zobrazení aktuální hodnoty. Nad stopami se nachází alfanumerický LCD displej 2*55 znaků. Vedle stop leží 9. motorizovaný potenciometr, který slouží jako hlavní zeslabovač. Nad tímto potenciometrem se nachází tlačítka pro výběr vlastnosti, kterou upravují rotační enkodéry v sekci stop. Zařízení dále obsahuje tlačítka a kolečko pro pohyb v nahrávce a sekci pro ovládnání přehrávání, resp. nahrávání. Poslední část kontroléru (vpravo nahoře) je věnována tlačítkům automatizace a filtrování výběru stop. V pravém horním rohu je LED displej zobrazující aktuální pozici/čas v nahrávce.

K propojení s počítačem jsou k dispozici dvě rozhraní. Prvním je klasické MIDI, druhým je USB-MIDI, které se po připojení k počítači tváří jako 4 virtuální MIDI porty. Zbýlé 3 porty, které nejsou využívány k ovládání kontroléru, jsou vyvedeny na zadní straně v podobě MIDI konektorů. Dále zde lze nalézt napájecí konektor pro napětí 7.5 V DC a vstupy pro dva nožní přepínače. Pro komunikaci s DAW je použit vlastní proprietární protokol *Mackie Control Protocol*, který funguje nad MIDI protokolem.



Obrázek 2.2: Mackie Control Unit Pro DAW kontrolér. Převzato z [7]

Ke kontroléru lze připojit až 3 další rozšiřující jednotky z rodiny Mackie MCU. Tato zařízení jsou vybavena pouze MIDI rozhraním a je doporučované k připojení využít porty, které nabízí popisovaný kontrolér. Výrobce nabízí dva typy rozšiřujících jednotek. První nese název *XT Pro* a nabízí 8 stop s motorizovanými potenciometry. Vzhledově vypadá stejně jako levá polovina kontroléru. Druhý typ jednotky nese označení *C4 Pro* a nabízí 32 rotačních enkodérů spolu se čtyřmi LCD displeji (opět 2*55 znaků). Pro připojení používá stejné rozhraní jako první jednotka. Tento typ jednotky je vhodný pro ovládání zvukových efektů a procesorů. Celkově je tedy možné po připojení rozšiřujících jednotek dosáhnout až 32 fyzických stop s potenciometry nebo 24 stop s ovladačem pro efekty.

Mackie Control Unit Pro je velmi povedeným zařízením. Cílem kontroléru je poskytnout uživateli pocit, jako kdyby pracoval s analogovým zařízením, což se vzhledem ke konstrukci zařízení a možnostem jeho rozšíření až na 32 fyzických stop povedlo. Cena zařízení 30 500 Kč (thonmann, 5. 1. 2019).

Solid State Logic Nucleus²

V textu je čerpáno z informací a materiálů uvedených na stránkách výrobce [15]. Vybrané zařízení od společnosti SSL je určeno do středně velkých nahrávacích studií. Oproti předchozímu kontroléru se jedná o větší zařízení, které v sobě navíc integruje zvukové rozhraní. Kontrolér má celkem 16 fyzických stop, přičemž rozložení ovládacích prvků stopy je podobné prvnímu popisovanému kontroléru. Oproti předchozímu má tento kontrolér navíc

indikaci signálu pro každou stopu tvořenou sloupcem 10-ti LED diod. Nad osmicí stop je umístěn alfanumerický LED displej 2*55 znaků. Zajímavý je střední panel, který je rozdělen na dvě části. V té spodní jsou umístěna tlačítka pro ovládání přehrávání a posun v nahrávce. V horní části jsou tlačítka pro přepínání virtuálního okna nad ovládanými zeslabovači a další tlačítka k ovládání DAW. Dále zde jsou pak prvky pro ovládání analogových vstupů a výstupů, které Nucleus nabízí. Zajímavostí jsou také tlačítka pro přepínání ovládaného DAW softwaru, jelikož tato konzole umí ovládat až 3 najednou.

Rozhraní této konzole je poněkud bohatší než v předchozím případě, a to díky tomu, že do sebe integruje více funkcionality. Zvukové rozhraní nabízí dva analogové vstupy s zesilovači SSL a efektovou smyčkou. Dále je zde pak několik výstupů pro sluchátka a hlavní monitory. Nucleus dále obsahuje USB rozhraní, přes které lze připojit k počítači a ovládat jím DAW software. Posledním důležitým rozhraním je Ethernetové, přičemž na zadním panelu jsou vyvedeny hned dvě zásuvky. Přes toto rozhraní je možné připojit konzoli k počítači a není tak nutné používat USB kabel. Zároveň je možné konzoli připojit do sítě *Dante*, což je rozhraní pro zvuk používané v profesionální sféře. Díky tomuto rozhraní lze ethernetovým kabelem přenášet nekomprimovaný zvuk na větší vzdálenosti bez vlivu elektromagnetického rušení. Po připojení konzole do této sítě jí lze ovládat připojená zařízení.



Obrázek 2.3: Konzole SSL Nucleus². Převzato z [15]

Ve své kategorii se jedná o zařízení, které nemá konkurenci. Konzole kombinuje analogové rozhraní s digitálním rozhraním Dante a možností ovládání až 3 DAW s podporou protokolů HUI a MCU. Díky podpoře těchto dvou protokolů jím lze ovládat většinu nahrávacích programů na trhu. Vzhledem ke všem možnostem, které nabízí, je konzole určena především do větších středních studií a jiných prostředí, kde se pracuje se zvukem – např. filmová postprodukce. Cena zařízení je 118 000 Kč (thonmann, 5. 1. 2019).

Presonus FaderPort

V textu je čerpáno z informací a materiálů uvedených na stránkách výrobce [12]. Třetí řešení se poněkud liší od předchozích dvou, která byla zaměřena především na profesionální uživatele ve středně velkých studiích. Popisované zařízení nalezne své uplatnění hlavně v domácích a mobilních podmínkách. Oproti předchozím disponuje pouze jedním motorizovaným potenciometrem, kterým však lze díky přepínání mezi virtuálními stopami ovládat všechny stopy v DAW. K ovládané stopě jsou dále vázána tlačítka MUTE a SOLO. Na kon-

troléru lze dále nalézt ovládací prvky pro posun v nahrávce (rotační enkodér) a tlačítka pro ovládání přehrávání, resp. nahrávání. Nakonec jsou zde prvky pro ovládání automatizace a vlastností připojeného DAW. Všechna tlačítka mají přiřazenu druhou funkci, která je aktivována klávesou *SHIFT*.

Zařízení poskytuje nativní podporu pro nahrávací software Studio One, který je z dílny stejnojmenné firmy, dále pak podporuje protokoly HUI a umí emulovat Mackie Control Universal zařízení – popisované výše. Díky tomu jej lze připojit a ovládat s ním téměř libovolný z dostupných DAW softwarů. Rozlišení potenciometru je 10 bitů, což dává 1024 kroků na 100 mm dráhu jezdce.



Obrázek 2.4: Presonus FaderPort. Převzato z [12]

Rozhraní kontroléru je taktéž velmi jednoduché. K počítači jej lze připojit pomocí USB, pro napájení je použit 12V 2A zdroj. Na zadní straně se dále nachází vstup pro nožní spínač, který umožňuje ovládat spuštění a zastavení přehrávání bez použití rukou.

I přes jeho malou velikost a omezený počet ovládacích prvků lze tímto zařízením téměř plnohodnotně ovládat celý nahrávací software. Právě kompaktnost dělá z tohoto zařízení ideálního pomocníka do malých i domácích studií, kde se většina zpracování zvuku provádí v softwaru. Cena zařízení je 4850 Kč (thonmann, 5. 1. 2018).

2.4 Zhodnocení

Cílem DAW kontroléru je dát uživateli pocit, jako by pracoval s analogovým mixážním pultem a navíc mu zprostředkovat důležité funkce digitálního nahrávacího prostředí bez použití myši a klávesnice. Na ovládací konzoli, na rozdíl od klasického analogového pultu, nejde tolik o celkový počet fyzických stop se zeslabovači, jelikož je zde možnost volby, které stopy virtuálního mixu v DAW softwaru budou ovládány fyzickými prvky na kontroléru. Tato funkce je typicky implementována jako okno nad virtuálním mixem, které lze nad stopami

v něm posouvat o 1 stopu nebo o počet fyzických stop kontroléru. Typické počty stop kontrolérů jsou 8 nebo 16, přičemž některá zařízení lze rozšířit o moduly s dalšími stopami.

Ovládací prvky všech popisovaných zařízení lze rozdělit do 3 skupin:

- Sekce stop
- Navigace v nahrávce a ovládání přehrávání
- Automatizace a nástroje DAW

Sekce se stopami má za cíl vytvářet pocit analogového zařízení, klíčovou roli zde hraje motorizovaný tahový potenciometr. Jeho rozlišení je většinou 10 bitů. Místo několika otočných potenciometrů, které jsou typické pro analogový pult, je použit jeden rotační enkodér s tím, že ovládanou vlastnost lze vybrat. V této sekci se dále nachází několik tlačítek pro každou stopu, které mají opět paralelu v analogovém světě. Výhodou digitálního zařízení je displej, na kterém může být uveden název stopy nebo hodnota upravované vlastnosti.

Sekce s navigací v nahrávce obsahuje rotační enkodér pro pohyb na časové ose nahrávky. Pohybovat se lze i čtveřicí tlačítek nahoru/dolů/vlevo/vpravo. Součástí jsou i tlačítka pro vkládání značek na časovou osu. Pro ovládání přehrávání slouží pětice tlačítek, podobná těm, které bylo dříve možné nalézt na kontroléru k vícestopému magnetofonu. Jedná se o tlačítka přehrát, stop, nahrávat, převíjení doleva/doprava.

Automatizační sekce se zaměřuje na ovládání funkcí digitální nahrávací stanice. Ovládací prvky této sekce lze rozdělit do několika skupin. Do jedné lze zařadit prvky pro ovládání automatizace – zapínání, resp. vypínání pro určité parametry. Další skupina slouží k vyvolávání nejčastějších příkazů DAW. Na vyspělejších kontrolérech dokonce existuje skupina, jejímž ovládacím prvkům může uživatel přiřadit některou z funkcí a přizpůsobit si tak kontrolér. Důležitou skupinou je výběr vlastnosti ovládané rotačními enkodéry v sekci stop. Na některých kontrolérech se jedná o tlačítka, u jiných o displej s dalším rotačním enkodérem nebo tlačítky.

Z případových studií vyplývá, že připojení k počítači je realizováno přes rozhraní MIDI, resp. USB/MIDI nebo ethernet s dodatečným softwarem v počítači. K přenosu dat jsou většinou použity proprietární protokoly – pro ovládání produktů výrobce kontroléru. Většina kontrolérů se však snaží být kompatibilní s co nejvíce nahrávacími softwary, a proto se snaží emulovat protokoly jiných stanic. Nejvíce emulované jsou Mackie HUI a novější Mackie Control Protocol, které mají podporu ve většině DAW.

Zajímavou vlastností je možnost rozšířit kontrolér o modul s další sekci stop. Tímto způsobem lze jednoduše zvýšit počet stop, které lze ovládat najednou a přiblížit se tak o krok blíže k uživatelskému pocitu, že pracuje s analogovým mixážním pultem. Toto je možné u prvního z popisovaných kontrolérů, ovšem na trhu lze nalézt i několik jiných kontrolérů, které mají tuto vlastnost.

Kapitola 3

Přenos dat mezi kontrolérem a PC

Z předchozí kapitoly vyplývá, že většina způsobů komunikace s DAW je proprietárními protokoly. V této práci budeme vycházet z dokumentace k DAW kontroléru *Logic Pro Dedicated Control Surface*, ve kterém je způsob komunikace popsán. Ve většině protokolů (včetně proprietárních) pro komunikaci tohoto typu je jako nižší vrstva použit protokol MIDI. Ten byl původně navržen pro přenos informací z klávesových nástrojů do syntezátorů, které vytvářely zvuk. Dnes je již hojně rozšířen, na jeho vývoji se podílí mnoho výrobců různých nástrojů a hudební elektroniky. Velká množství protokolů pro přenos dat do DAW softwaru používají právě tento protokol pro přenos dat, přičemž interpretace zpráv je jiná než v případě klasického použití. V této kapitole bude nejprve probrán protokol MIDI a dále pak modifikace tohoto protokolu pro účely výměny informací s kontrolérem. Na konci kapitoly bude rozebrán způsob připojení vytvářeného kontroléru k pracovní stanici.

3.1 MIDI

MIDI je zkratka pro Musical Instrument Digital Interface. Jedná se o průmyslový standard spravovaný organizací MMA (MIDI Manufacturers Association) – spojení mnoha výrobců zařízení, které MIDI používají. První verze protokolu byla zveřejněna v roce 1983. Standard specifikuje výměnu dat mezi různými zařízeními, a to jak na hardwarové, tak i softwarové vrstvě. Protokol byl původně navržen pro přenos informací mezi klávesovými nástroji a syntezátory, resp. zařízeními, která ukládají nebo vytváří zvuk. Z počátku bylo MIDI zaměřeno hlavně na přenos dat při živých vystoupeních, později však našlo uplatnění i v jiných aplikacích, jako například ovládání osvětlení nebo v nahrávacích studiích pro komunikaci s kontroléry. Jelikož se jedná o poměrně rozsáhlý standard, tato podkapitola se bude soustředit hlavně na ty pasáže, které budou později využity pro přenos informací z/do kontroléru. Tato podkapitola čerpá ze specifikace standardu MIDI [10].

MIDI protokol je určen pro přenos hudebních informací. Přenášen ovšem není digitalizovaný zvuk, ale pouze sled událostí, které nastávají na hudebním nástroji v reálném čase. Touto událostí může být například stisknutí, respektive uvolnění klávesy nebo změna nástroje. Výsledný zvuk je tvořen až v koncovém zařízení – syntezátoru.

Zprávy

V MIDI se komunikuje pomocí zasílání zpráv. Zpráva je typicky složena z více bajtů. Nejprve je poslán status byte, který je následován jedním až dvěma datovými bajty. Výjimkou jsou zprávy typu *Real-Time* a *Exclusive*. V rámci jednoho fyzického kanálu je možné přenést 16 logických kanálů.

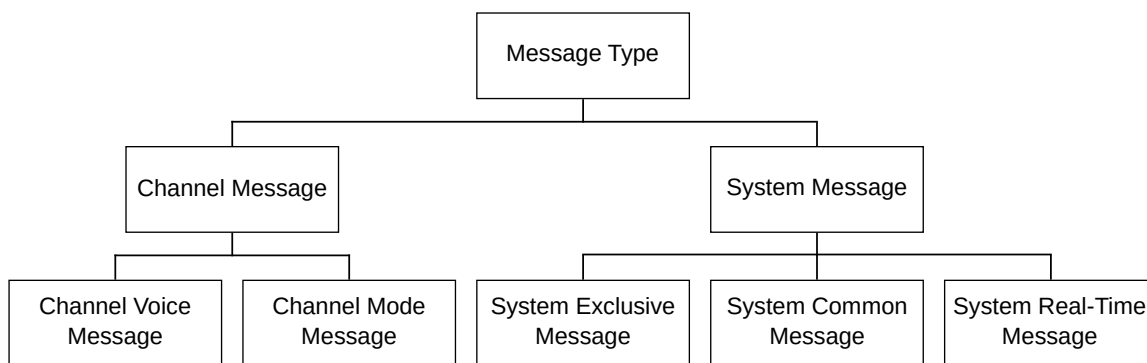
Protokol rozeznává pouze 2 druhy bajtů, které se od sebe liší nastavením nejvíce významového bitu (MSB):

- status byte – MSB je logická 0, rozsah 0x80 – 0xFF
- data byte – MSB je logická 1, rozsah 0x00 – 0x7F

Účelem status bajtu je identifikovat začátek a typ zprávy, aby bylo možné jednoznačně interpretovat následující datové bajty (pokud nějaké jsou). Zprávy jsou řazeny za sebou a nedochází k prokládání více zpráv do sebe. Jedinou výjimkou jsou zprávy typu *Real-Time*, které jsou tvořeny pouze status bytem a mohou být umístěny kdekoli v datovém toku. Pokud přijde nový status byte a předchodí zpráva nebyla dokončena, příjemce musí nedokončenou zprávu zahodit a přijímat novou.

Pro ušetření přenosového pásma byla vytvořena vlastnost *running status*. Cílem je neopakovat status bajty několika zpráv stejného typu, které následují těsně za sebou. Přenesen je pak pouze první status bajt, který je následován datovými bajty několika zpráv stejného typu. Running status končí, pokud je přijat nový status bajt. Výjimkou jsou opět zprávy typu *Real-Time*, které jej nepřerušují. Tato vlastnost je dostupná pouze pro zprávy typu *Voice* a *Mode*. Pokud příjemce přijme zprávu, pro kterou není implementovaná funkcionality v zařízení, zprávu ignoruje.

V MIDI protokolu jsou zprávy hierarchicky děleny do skupin. Na nejvyšší úrovni jsou rozděleny do dvou kategorií: *Channel Message* a *System Message*.



Obrázek 3.1: Hierarchie skupin MIDI zpráv. Převzato z [10]

Kanálové zprávy

Zprávy obsahující číslo logického kanálu. Jsou předurčeny pro posílání zpráv různým koncovým bodům, např. pokud hraje více nástrojů současně, každý používá vlastní kanál. Tento typ dělíme na dva podtypy:

- *Channel Voice Message* – zprávy ovládající zvuk nástroje
- *Channel Mode Message* – zprávy definující odezvu na zprávy typu *Voice Message*

Typ zprávy	Status byte	Počet datových bajtů
Note Off	0x8n	2
Note On	0x9n	2
Poly Key Pressure	0xA _n	2
Control Change	0xB _n	2
Program Change	0xC _n	1
Channel Pressure	0xD _n	1
Pitch Bend	0xE _n	2

Tabulka 3.1: Přehled zpráv typu *Channel Voice Message*. Parametr n ve sloupci *Status byte* udává číslo kanálu.

Zprávy typu *Channel Mode Message* oznamují připojenému sekvenceru, resp. MIDI zařízení, jak interpretovat příchozí zprávy typu *Channel Voice Message*. K tomu je využit status byte zprávy *Control Change*, přičemž první datový bajt je v rozmezí hodnot 0x78–0x7F. Hodnota datového bajtu určuje typ zprávy.

Zpráva *Note Off* je poměrně frekventovanou zprávou. Pro vylepšení efektivity při použití techniky Running Status je tato zpráva nahrazena zprávou *Note On* s použitím stejné hodnoty tónu (1. datový bajt) a nulové síly úderu (2. datový bajt). Obě zprávy mají ekvivalentní význam a měly by tak být chápány i připojeným zařízením.

Systémové zprávy

Zprávy typu *System Message* jsou určeny všem koncovým bodům a neobsahují tedy číslo kanálu. Lze je rozdělit na 3 podtypy:

- *System Exclusive Message* – zpráva, která může obsahovat libovolný počet datových bajtů. Tato zpráva je ukončena speciální sekvencí *EOX* (End of Exclusive) nebo příchozím status bajtem. Zpráva je uvozena identifikátorem výrobce. Pokud příjemce tomuto identifikátoru nerozumí, může zprávu ignorovat.
- *System Real-Time Message* – zprávy pro synchronizaci všech komponent systému, které obsahují synchronizovatelné hodiny. Zpráva se skládá pouze ze status bajtu a může se vyskytovat kdekoliv v proudu bajtů i uprostřed jiné zprávy.
- *System Common Message* – ostatní systémové zprávy určené pro všechny příjemce.

Typ zprávy	Podskupina	Status byte	Počet datových bajtů
System Exclusive	Exclusive	0xF0	∞
TimeCode Quarter Frame		0xF1	1
Song Position Pointer		0xF2	2
Song Select	Common	0xF3	1
Tune Request		0xF4	0
EOX (End of Exclusive)		0xF7	0
Timing Clock		0xF8	0
Start		0xFA	0
Continue	Real-Time	0xFB	0
Stop		0xFC	0
Active Sensing		0xFE	0
System Reset		0xFF	0

Tabulka 3.2: Přehled zpráv skupiny *System Message*

Hardware

Standard MIDI definuje i hardwarovou vrstvu, která je svými technickými možnostmi poplatná roku svého vzniku. Nutno dodat, že v hudební praxi je stále často používána, avšak v poslední době vytlačována USB rozhraním, do kterého jsou MIDI zprávy zapouzdřeny.

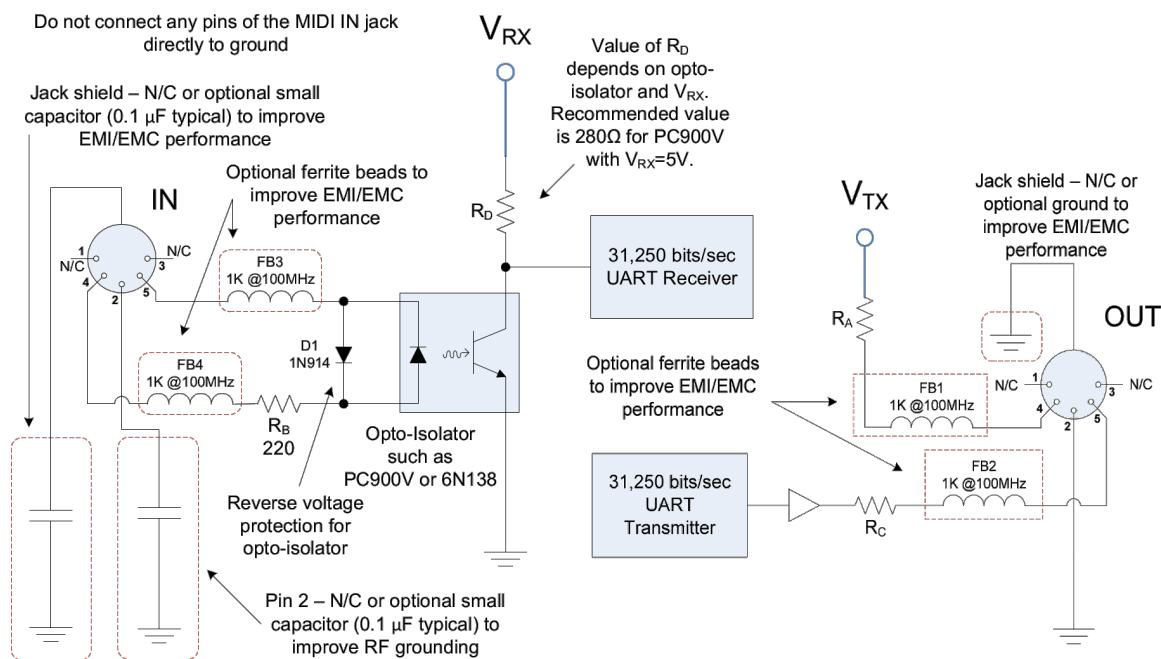
MIDI rozhraní, tak jak jej definuje standard, je klasické sériové asynchronní rozhraní provozované na rychlosti 31.25 kBaud ($\pm 1\%$) s jedním start bitem, 8 datovými bity a jedním stop bitem. Pořadí odesílaných bitů bajtu je LSB first (nejdříve méně významové bity). Start bit má hodnotu logické 0, stop bit logické 1.

Obvodově se jedná o proudovou smyčku s proudem 5 mA. Proud smyčkou prochází při logické 0. Aby nedocházelo k zemnicím smyčkám, vysílač a přijímač je oddělen optočlenem.

Původní návrh elektrické specifikace z roku 1996 [10] počítá s napětím 5 V, které bylo v té době běžně využíváno v rámci TTL logiky. Roku 2014 vyšla nová revize elektrické specifikace [9], která počítá s napětími 3.3 V a navíc zavádí RF a EMI vylepšení.

3.2 Komunikační protokol

Jak již bylo naznačeno v úvodu kapitoly, pro komunikaci mezi kontrolérem a DAW softwarem je ve většině případů používán MIDI protokol. Tento protokol ovšem přímo nespecifikuje příkazy pro ovládání DAW kontrolérů, a proto jsou použity zprávy, které protokol nabízí, ovšem v jiném významu než definuje standard. Na trhu existuje mnoho DAW kontrolérů, bohužel však neexistuje jednotný způsob komunikace s DAW softwarem. Navíc podpora kontrolérů není úplná ani ze strany DAW softwaremů. Proto se v rámci této práce zaměříme na studium již existujících způsobů komunikace. Jako referenční nahrávací software bude použit *Steinberg Cubase Elements* verze 8.



Obrázek 3.2: Standardní zapojení MIDI rozhraní dle novější revize. Ze schématu jsou patrné konektory DIN5 pro vstup, resp. výstup a oddělení optoizolátorem na straně příjemce. Převzato z [9], upraveno

Pro komunikaci s kontrolérem lze použít novější protokol využívaný zařízením *Logic Pro Dedicated Control Surface*, v jehož manuálu [3] je protokol popsán. Tento protokol však není podporován vybraným referenčním softwarem, proto bude použit starší protokol *Mackie HUI*. Rozdíl mezi těmito protokoly je pouze v jiných typech použitých MIDI zpráv. Tato podkapitola bude vycházet ze zdrojů [11] a [8]. Jejím cílem je uvést stručný přehled vyměňovaných zpráv mezi nahrávací stanicí a kontrolérem.

System Exclusive zprávy

V protokolu je často využito MIDI zpráv typu System Exclusive, tedy zpráv s libovolnou délkou datové části. Pro zjednodušení nahradíme hlavičku této zprávy značkou <hdr>. Hlavička se skládá z pětice bajtů: F7 00 00 66 05 00. V této posloupnosti je zakódován začátek zprávy a identifikace zařízení.

Tahové potenciometry

Přijato:	B0 0i hh 2i 1l	Povel k posunu od DAW
Odesláno:	B0 0i hh 2i 1l	Uživatel posunul potenciometrem

Formát zprávy je stejný pro příjem i odeslání. Návěští i reprezentuje identifikátor potenciometru: 0–7. Hodnota pozice je 14-ti bitová, přičemž je rozdělena po sedmi bitech do dvou datových bajtů. Nejprve je přenesena horní polovina (hh), poté spodní (1l). Kontroléry typicky nepracují s přesností na 14 bitů, proto hodnotu před odesláním, resp. po přijetím, musí upravit přidáním, resp. odebráním nulových bitů zprava.

Tlačítka

Přijato: –
Odesláno: B0 0F ii 2C sp Stisk/uvolnění tlačítka nebo doteku potenciometru

Tato zpráva je odeslána kontrolérem, pokud bylo stisknuto, resp. uvolněno tlačítko nebo došlo k doteku, resp. uvolnění jezdce potenciometru. Návěští *ii* spolu s návěštím *p* reprezentují adresu tlačítka, přesné mapování adres lze dohledat v příloze dokumentu, ze kterého podkapitola vychází. Nibble *s* s návěštím (*s*) nese informaci o stisku: 0x4 = stisk/dotek, 0x0 = uvolnění.

Led diody

Přijato: B0 0C ii 2C sp Povel k rozsvícení/zhasnutí
Odesláno: –

Většina tlačítek kontroléru je podsvětlená. Pokud je od DAW přijata tato zpráva, jedná se o pokyn k rozsvícení nebo zhasnutí LED diody příslušného tlačítka. Datový bajt *ii* spolu s návěštím *p* je stejně jako u předchozí zprávy adresou tlačítka, u kterého operujeme s podsvětlením. Hodnota nibblu *s* určuje akci: 0x4 = rozsvícení, 0x0 = zhasnutí.

Rotační enkodéry

Přijato: –
Odesláno: B0 1i XX Zpráva odeslaná při otočení enkodéru

Při otočení rotačního enkodéru odešle kontrolér zprávu se smyslem otáčení a počtem zarážek, o které byl otočen. Z konstrukčního hlediska lze enkodérem otáčet o 360°, přičemž za jednu otáčku projde typicky 24 zarážek (liší se podle typu enkodéru). Návěští *i* značí identifikaci stopy, ve které se enkodér nachází, v rozmezí 0x00 – 0x07. Druhý datový bajt (*XX*) má následující bitový formát: (0 *s* *v* *v* *v* *v* *v* *v*). Smysl otáčení je určen bitem *s*: 0 = po směru hodinových ručiček, 1 = proti směru hod. ručiček. Hodnota *vvvvv* určuje počet zarážek, o které byl enkodér otočen. V zařízení jsou použity rotační enkodéry s tlačítkem, stisk enkodéru je přenášen zprávou pro stisk tlačítka uvedenou výše.

Indikace natočení enkodérů

Přijato: B0 1i XX Změna hodnoty natočení ve stanici
Odesláno: –

Každý z enkodérů je obklopen 12-ti LED diodami, které indikují úroveň jeho natočení. Tato zpráva je odeslána stanicí při každé změně. Návěští *i* má stejný význam jako u předchozích zpráv: identifikace stopy, ve které se indikátor nachází. Bajt *XX* nese informaci o rozsvícených diodách. Jeho podoba je kompozicí několika hodnot: (0 *p* *x* *x* *v* *v* *v* *v*). Bit *p* určuje, zda je rozsvícena LED dioda umístěná pod enkodérem. Hodnota *xx* určuje jeden ze čtyř módů svícení. Bity *vvvv* určují, které LED diody mají být rozsvícené. Konkrétní mapování LED diod v závislosti na použitém módu lze nalézt příloze zdroje [11].

Enkodér pozice

Přijato: BO OD XX
Odesláno: –

Rotační enkodér pro pohyb v nahrávce se nachází v navigační sekci kontroléru. Zpráva je odeslána kontrolérem vždy při otočení enkodérem. Datový bajt XX svým tvarem odpovídá stejnojmennému bajtu u zprávy pro rotační enkodéry.

Hlavní LCD displej

Přijato: <hdr> 12 oo yy ... F7
Odesláno: –

Tato zpráva zapisuje znaky na hlavní displej umístěný nad sekci stop. Jeho velikost je 2x40 znaků. Na každou stopu je tedy k dispozici 5 znaků v každém řádku. Interně je displej rozdělen na 8 zón, které mohou být přeneseny současně nebo každá zvlášť, případně v libovolné kombinaci. Návěští oo určuje přenášenou zónu (0x00 – 0x07). Za ním následuje 10 datových bajtů se znaky k zobrazení. Těchto 11 bajtů (návěští + datové bajty) se může několikrát opakovat za sebou pro docílení přenosu více zón současně. Celá zpráva končí bajtem s hodnotou 0x7F.

Měření úrovně signálu

Přijato: A0 0i sv Hodnota aktuální síly signálu pro zobrazení na ledbaru
Odesláno: –

K měření úrovně signálu příslušné stopy je použit sloupec 13-ti LED diod. Rozsvícený sloupec diod zobrazuje aktuální špičku signálu na dané stopě. Kontrolér tyto zprávy pouze přijímá a ovládá LED diody dle instrukcí.

Nibble s návěštím i adresuje stopu (hodnoty 0x0 – 0x7). Návěští s udává stranu, pro kterou je zaslána hodnota platná: 0x0 pro levou stranu, 0x1 pro pravou. Nibble s návěštím v udává hodnotu, kolik LED diod má být rozsvícených v rozmezí od hodnoty 0x0 (všechny LED diody zhasnuté) až po hodnotu 0xC (všechny LED diody rozsvícené).

3.3 Spojení se stanicí

V předchozím textu byl představen protokol a MIDI vrstva, kterou lze využít pro komunikaci se stanicí. V této podkapitole se budeme zabývat spojením kontroléru a audio stanice na nižší vrstvě, jejímž cílem je přenos posloupnosti bajtů MIDI zpráv. K dispozici je několik způsobů. Nejpřímočařejším je použít k přenosu hardwarovou MIDI vrstvu tak, jak byla definována ve standardu. Většina moderních DAW kontrolérů však již toto rozhraní nenabízí a jediným rozhraním je USB. Další alternativou, která je v hudební branži stále populárnější, je přenos dat přes ethernetové rozhraní. Na straně počítače je vždy vytvořen MIDI port, ať již jde o skutečný hardwarový nebo virtuální, v případě použití USB nebo Ethernetu. Pro audio stanici je tedy způsob fyzického propojení počítače s kontrolérem zcela transparentní. První z možností byla rozebrána v předchozím textu, dvě alternativní možnosti budou představeny nyní.

USB-MIDI

Použití klasického MIDI rozhraní je dnes již zastaralé a jeho přenosové rychlosti neodpovídají požadovaným rychlostem. Také vzhledem k miniaturizaci počítačů a sjednocení rozhraní je výhodné používat USB, jelikož MIDI rozhraní již není standardní součástí zvukových karet a je třeba dokoupit speciální hardware. Organizace *The USB Implement Forum* vydala specifikaci pro připojení MIDI zařízení přes USB rozhraní k počítači [4]. Z tohoto dokumentu budou vycházet následující odstavce. Rozhraní je nazýváno USB-MIDI, jelikož není hardwarově kompatibilní s řešením, které je definováno v MIDI standardu. USB-MIDI rozhraní je částí třídy USB audio zařízení. Díky tomu může zařízení pracovat s ovladači, které jsou již součástí operačního systému (tzv. „Class Compliant“), pokud nemá zařízení specifické požadavky – v tom případě je nutné, aby výrobce dodal vlastní proprietární ovladače.

USB-MIDI posouvá tradiční rozhraní MIDI dále, zlepšuje jeho přenosovou rychlost a umožňuje plně využít všech 16 MIDI kanálů při plném zatížení. Navíc zavádí funkci virtuálních kabelů, kdy jedním USB-MIDI rozhraním lze přenést až 16 virtuálních rozhraní. Dohromady lze tedy jedním USB-MIDI rozhraním přenést až 256 kanálů. Datové toky jednotlivých virtuálních kabelů jsou na straně odesílatele multiplexovány do jednoho proudu dat, přeneseny jednou USB rourou a na straně příjemce opět demultiplexovány na jednotlivé MIDI toky. Data jsou posílána ve formě tzv. USB-MIDI paketů nesoucích MIDI zprávu. Tyto pakety mají pevnou velikost 4 B, což je efektivní vzhledem k tomu, že velikost většiny MIDI zpráv je do 3 B. První bajt paketu slouží k identifikaci virtuálního kabelu a signalizaci přenášených dat.

Byte 0		Byte 1	Byte 2	Byte 3
Cable Number	Code Index Number	MIDI_0	MIDI_1	MIDI_2

Obrázek 3.3: Struktura USB-MIDI paketu. Převzato z [4]

V rámci USB protokolu je použit přenos typu bulk. Pro komunikaci nabízí USB-MIDI specifikace rozhraní *MIDIStreaming*. Pro základní komunikaci se předpokládají dva USB endpointy: jeden pro komunikaci od zařízení k počítači (IN), druhý od počítače k zařízení (OUT). Specifikace dovoluje využít i více endpointů pro odlišení toků, nicméně kvůli efektivitě doporučuje využít jeden endpoint pro každý směr s využitím funkce virtuálních kabelů. Přenos dat pak probíhá následovně: na straně odesílatele jsou shromážděny MIDI zprávy od všech odesílajících aplikací. Následně proběhne překlad zpráv do formy USB-MIDI paketů a ty jsou multiplexovány do jednoho datového toku, který je odeslán příjemci. Příjemce demultiplexuje přijatý tok dat, podle označení jednotlivých paketů zrekonstruuje původní MIDI zprávy a ty distribuuje příslušným MIDI portům, kde čekají na data různé aplikace. Přenos obráceným směrem je totožný s rozdílem použití jiného USB endpointu.

RTP-MIDI

Ethernet je v audio-technice využíván stále častěji. Kromě přenosu digitálního zvuku na větší vzdálenost (např. Dante) jej lze použít i k výměně MIDI zpráv. Ty jsou přenášeny pomocí RTP protokolu (Real-time Transport Protocol), který je běžně používán pro přenos zvuku nebo videa, avšak jeho formát umožňuje přidávat nové kodeky a této skutečnosti bylo využito při návrhu RTP-MIDI. Následující odstavce budou vycházet z dokumentu RFC 6295 [6] a článku [16].

RTP-MIDI je otevřený IETF standard, který umožňuje přijímat, resp. odesílat standardní MIDI 1.0 zprávy přes síťové prostředí. Protokol je navržen robustně, aby jej bylo možné použít jak při studiové práci, tak i při živých vystoupeních. RTP-MIDI obsahuje mechanismus pro obnovu dat (Midi event journaling), který snižuje latenci v případě poškození dat při přepravě: není nutné znovu-vyslání odesílatelem. RTP může být přenášeno unicastově nebo multicastem s použitím vrstev TCP nebo UDP. Výběrem vhodné vrstvy lze docílit bezchybného přenosu dat nebo kratší latence. V případě použití spolehlivé vrstvy TCP zůstane interní mechanismus obnovy dat nepoužitý.

Přenos tímto protokolem lze realizovat ve všech majoritních operačních systémech (Linux/Mac/Android/Windows). V počítači je nutné mít nainstalován speciální software, který bude přijímat data na určené IP adrese a portu. Při startu tohoto programu je vytvořen virtuální MIDI port, se kterým budou komunikovat aplikace vyžadující MIDI rozhraní. Cílem softwaru je pak transformovat přijatá data do formy MIDI zpráv a ty posílat na příslušný MIDI port. Protokol je navíc dostatečně malý (prostorově i výpočetně) na to, aby mohl být implementován v dnešních vestavěných systémech, jako například klávesových nástrojích, kontrolérech atd.

Podobně jako u předchozího případu, RTP-MIDI není náhradou MIDI 1.0 protokolu, ale rozšířením o rychlejší a spolehlivější přenos MIDI zpráv. RTP-MIDI protokol je obzvláště vhodný pro aplikace, které komunikují prostřednictvím MIDI na větší vzdálenost (například v nahrávacím studiu nebo na pódiu).

Kapitola 4

Návrh architektury zařízení

Tato kapitola se bude nejprve věnovat konstrukci vybraného motorizovaného potenciometru, který je ústředním prvkem uživatelského rozhraní kontroléru. Následně bude popsán návrh architektury vytvářeného zařízení. Jedná se o modulární architekturu, proto bude nejdříve provedeno rozdělení zodpovědností mezi jednotlivé moduly, následně bude popsáno spojení modulů dohromady a jejich kooperace. Po představení architektury z vyšší úrovně abstrakce se bude kapitola zabývat návrhem architektur jednotlivých modulů.

Jelikož se jedná o poměrně komplexní zařízení, které pracuje v reálném čase, je vhodné delegovat zodpovědnosti mezi jednotlivé funkční celky – moduly. Tento krok je výhodný nejen z hlediska vývoje a testování, ale i z hlediska případných oprav a rozšiřitelnosti zařízení v budoucnu.

4.1 Motorizovaný potenciometr

V této kapitole bude rozebrán motorizovaný potenciometr z hlediska jeho funkce v zařízení, konstrukce a způsobu snímání, resp. ovládání. Jak vyplývá z předchozích kapitol, motorizovaný posuvný potenciometr je ústředním prvkem uživatelského rozhraní zařízení. Jeho přesné a precizní ovládání je naprosto klíčové pro správnou funkci kontroléru.

Na trhu existuje pouze několik renomovaných výrobců, kteří se zabývají výrobou motorizovaných potenciometrů. Hlavním kritériem při jeho výběru byla spolehlivost a montážní výška pro zabudování do zařízení. Vybraným potenciometrem je *ALPS RSA0N11M9A0K*. Hlavním důvodem pro tento potenciometr byla kromě spolehlivosti jeho výška, která je velmi nízká díky horizontálnímu umístění motoru pod samotný potenciometr. Tyto potenciometry jsou často využívány v komerčně úspěšných digitálních pultech a DAW kontrolérech.



Obrázek 4.1: Motorizovaný potenciometr ALPS RSA0N11M9A0K. Zdroj [5]

parametr	hodnota
posuvná vzdálenost	100 mm
charakteristika	lineární
hodnota odporu	10 k Ω
odchylka odporu	$\pm 20\%$
odpor dotykové dráhy	max 1 k Ω
operační teplota	-10 °C – 60 °C
životnost	30 000 cyklů
napětí motoru	10 V DC
maximální proud @10 V	800 mA
zatížitelnost	0.5 W

Tabulka 4.1: Vybrané parametry zvoleného potenciometru. Přejato z [1]

Konstrukce

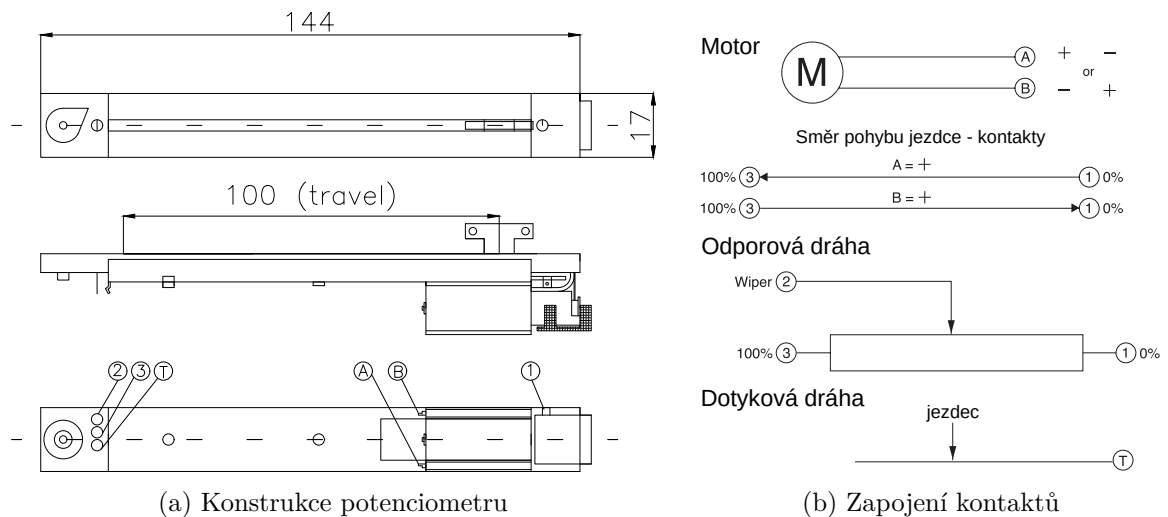
Tato podkapitola vychází z materiálů dostupných na stránce produktu [1]. Motorizovaný potenciometr má poněkud složitější konstrukci, než je tomu u klasického posuvného potenciometru v analogovém mixážním pultu, kde se nachází pouze jedna odporová dráha (případně i více). Konstrukčně jej lze rozlišit 3 samostatně funkční celky:

- odporová dráha
- snímač dotyku
- motor

Odporová dráha je společným prvkem s klasickým posuvným potenciometrem. Na rozdíl od analogového pultu, kde potenciometrem prochází zeslabovaný zvukový signál, v digitálním kontroléru snímáme pozici jezdce na dráze. Potenciometr je v obvodu zapojen jako dělič napětí. Z toho vyplývá i jeho charakteristika, která je lineární, na rozdíl od analogu, kde bývá použita logaritmická. Poloha jezdce potenciometru se jeví jako analogová hodnota v rozmezí 0 V až referenční napětí. Pro její snímání používáme AD převodník.

Snímač dotyku je konstrukčně řešený jako samostatná dráha podél odporové dráhy. Jedná se o kapacitní dotykový snímač s jedním vývodem, který se připojí k integrovanému obvodu určenému ke snímání dotyku. S tímto obvodem je pak možné komunikovat pomocí některé z klasických sběrnic, kde dostáváme binární hodnotu, zda je snímač dotknut. Snímání dotyku je důležité hned ze dvou důvodů. První souvisí s automatizací v prostředí DAW, kde dotek potenciometru způsobí začátek, resp. konec nahrávání automatizace zeslabovače. Druhá funkce je ochranná, kdy je třeba zajistit, aby v momentě dotyku zeslabovače byl motor vypnutý, uživatel netlačil do rozběhnutého motoru a nedošlo k přetržení řemene.

Nedílnou součástí motorizovaného potenciometru je samotný motor. Jde o malý stejnosměrný motorek, který je řemenem pevně spojen s jezdce potenciometru. Při ovládání je třeba dbát na dojezd jezdce do krajních poloh, jelikož zde není žádná ochrana. Při trvale zapnutém motoru by tak mohlo dojít k jeho přetržení.

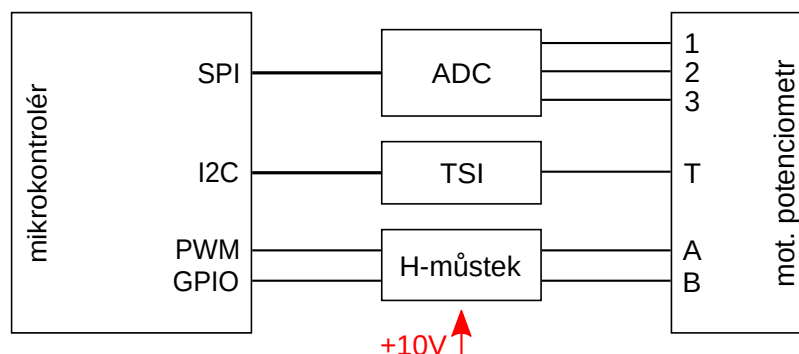


Obrázek 4.2: Konstrukce a zapojení kontaktů vybraného motorizovaného potenciometru. Přejato [1], upraveno

Řízení DC motoru

Jelikož se jedná o klasický DC motorek, nelze pouze ze signálů řídicích motor zjistit, o kolik otáček se motor otočil. Pro řízení motoru je tedy nutná nejprve precizní implementace zjišťování polohy jezdce a jeho dotyku.

Polohu jezdce zjistíme AD převodníkem s použitím zapojení uvedeném v textu výše. Uvnitř modulu, který řídí potenciometr se bude pracovat v rozlišení 16 bitů. Vně modulu pak v rozlišení 14 bitů, případně v nižším, pokud to bude vyžadovat použitý komunikační protokol. Vzorkovací frekvence převodníku by měla být dimenzována úměrně k rychlosti, kterou se jezdec pohybuje. Při maximální rychlosti jezdce dojde ke změně o 1 dílek při použití 16-ti bitového rozlišení zhruba za 0.0076 ms. Pokud bychom chtěli zaznamenat každou změnu, musela by vzorkovací frekvence být přibližně 131 570 Hz. V praxi však budou otáčky motoru řízeny a při dojíždění do chtěné polohy bude zpomalovat, proto bude možné použít převodník s nižší vzorkovací frekvencí.



Obrázek 4.3: Blokový náčrt zapojení pro řízení motoru potenciometru

Směr otáčení motoru je určen polaritou napájení přivedeného na svorky. Rychlost otáčení je úměrná velikosti napájení. Pro řízení otáček motoru je vhodné použít H-můstek spolu s PWM regulací. H-můstek je sada výkonových tranzistorů, typicky uzavřená v 1 pouzdře, která má za cíl s pomocí malých proudů a malého napětí od mikrokontroléru ovládat vyšší napětí s vyššími proudy, které je přivedeno k motoru. S pomocí vhodného zapojení tranzistorů, označovaného jako „full-bridge“, lze dosáhnout i možnosti ovládat polaritu napájení na motoru a tím i smysl otáčení.

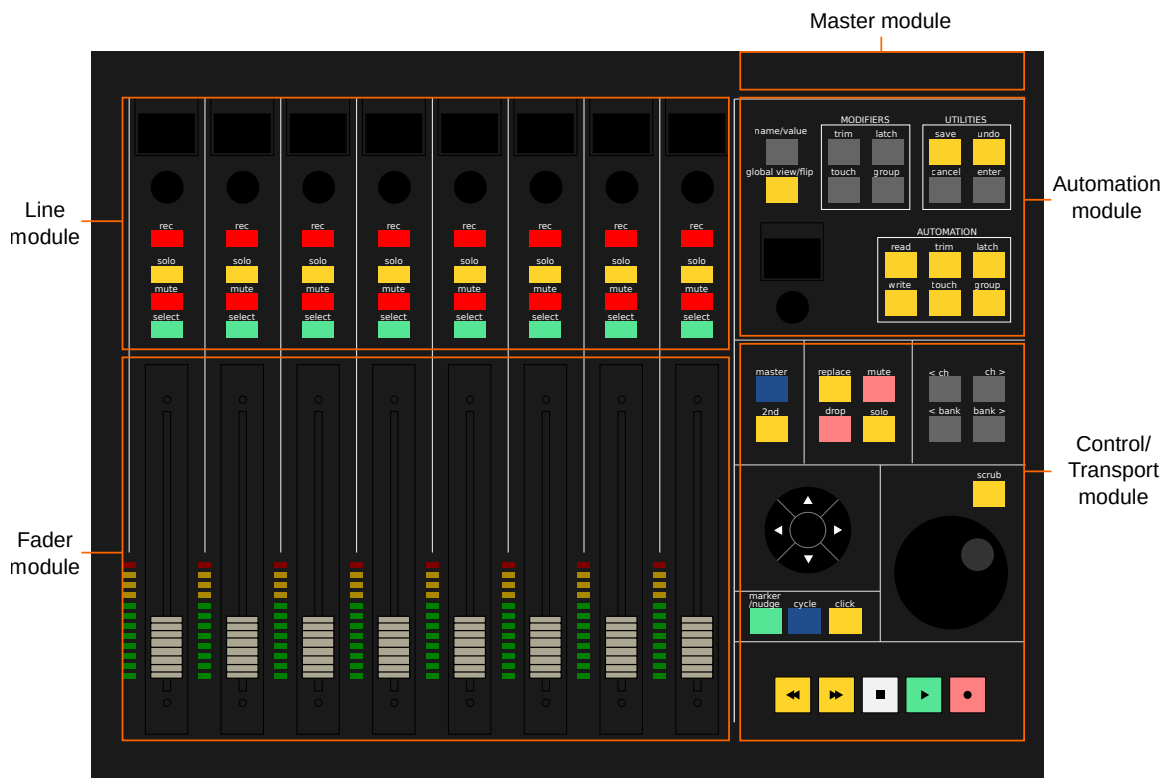
S pomocí pulzně šířkové modulace (PWM) lze dosáhnout řízení otáček motoru. PWM signál je přiveden na vstup H-můstku, který ovládá výkonové tranzistory. Druhým vstupem je signál určující směr motoru.

4.2 Rozložení ovládacích prvků

Rozložení ovládacích prvků kontroléru vychází z rozebraných řešení v kapitole 2. V levé části navrhovaného zařízení se nachází sekce s osmi fyzickými stopami. Jedna stopa se skládá z následujících ovládacích prvků: motorizovaného potenciometru, sloupce LED diod pro indikaci síly signálu, tlačítek pro výběr stopy (**Select**), nahrávání (**Rec**), zeslabení (**Mute**) a samostatného přehrávání (**Solo**). Dále pak z rotačního enkodéru s tlačítkem a OLED displeje. Pro ušetření místa byla funkcionality hlavního zeslabovače (tedy devátého motorizovaného potenciometru) spojena s osmým potenciometrem a bude možné mezi oběma funkcionalitami přepínat tlačítkem.

V pravé třetině kontroléru se nachází sekce pro ovládání přehrávání. Ovládací prvky budou popisovány odspoda. Nejnižše se nachází tlačítka pro ovládání přehrávání: spuštění, pozastavení, nahrávání a převíjení oběma směry. Nad nimi se nachází tlačítka pro ovládání metronomu, cyklení ve výběru a vložení výběrové značky. Směrem výše následuje rotační enkodér s velkým ovládacím prvkem pro jemný horizontální posun v nahrávce a tlačítkem **Scrub** pro přehrání části nahrávky při otáčení enkodérem. Vlevo od enkodéru se nachází čtyři tlačítka uspořádaná do kruhového navigačního prvku. Ten slouží k horizontálnímu i vertikálnímu pohybu v nahraných stopách na obrazovce audio stanice. Nad těmito prvky se nachází dvě řady tlačítek. Tlačítko **Master** slouží k přepínání funkcionality osmé stopy a hlavního zeslabovače, tlačítko **2nd** přepíná na alternativní funkcionality ostatních tlačítek. Dále následuje skupina prvků pro ovládání chování DAW při nahrávání (tlačítka **Replace**, **Drop**, **Mute**, **Solo**). Následující skupina tlačítek posouvá okno ovládaných stop v rámci všech stop virtuálního mixéru audio stanice. Posun je možný oběma směry, přičemž tlačítka označená jako **<Bank** a **Bank >** posouvají o počet fyzických stop kontroléru, tlačítka nad nimi (**<ch** a **ch >**) lze okno posunout o jednu stopu.

V pravém horním rohu se nachází displej s rotačním enkodérem, který primárně slouží k výběru vlastnosti ovládané rotačními enkodéry v sekci stop, sekundárně pak k nastavení celého kontroléru. Nad displejem se nachází tlačítko **Flip**, které přenesse funkcionality rotačních enkodérů stop na motorizované potenciometry. Díky tomu má uživatel jemnější kontrolu nad nastavovanou hodnotou. Nad ním se nachází prvek **Name/Value**, který přepíná zobrazení na displejích v sekci stop. Dále pak následují tři skupiny tlačítek. Skupina *Automation* ovládá automatizaci audio stanice: tlačítka **Read**, **Write**, **Trim**, **Touch**, **Latch** a **Group**. Skupina *Utilities* jsou často používaná tlačítka pro ovládání nahrávacího programu. Jde o tlačítka pro uložení (**Save**), zrušení (**Cancel**), potvrzení (**Enter**) a tlačítka vrácení se o krok zpět (**Undo**). Poslední skupina *Modifiers* obsahuje tlačítka pro přepínání módů kontroléru a uživatelsky programovatelná tlačítka.



Obrázek 4.4: Návrh rozložení horního panelu kontroléru s rozdělením ovládacích prvků do jednotlivých modulů (oranžové ohraničení)

Rozhraní pro připojení k počítači s audio stanicí bude umístěno na zadním panelu. Jelikož se jedná o zařízení určené pro výzkumnou činnost, bude obsahovat všechny tři rozhraní popsaná v kapitole 3. Kontrolér tedy bude obsahovat jak staré rozhraní MIDI, které bude díky své jednoduchosti sloužit jako odrazový můstek při referenční implementaci firmwaru, tak i moderní rozhraní USB-MIDI a RTP-MIDI, na kterých bude možné provádět výzkumnou činnost.

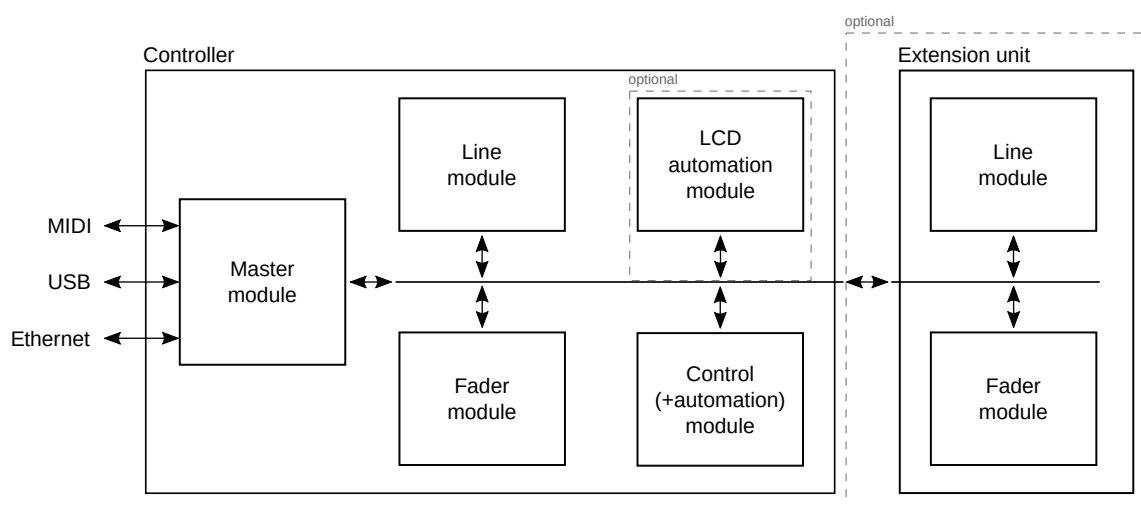
4.3 Rozdělení do modulů

Rozdělení na jednotlivé moduly zhruba odpovídá sekcím uvedeným ve zhodnocení kapitoly 2. Při rozdělování byla zohledněna i konstrukční stránka zařízení z hlediska hloubky usazení DPS pod vrchním panelem, jelikož různé ovládací prvky mají různou výšku a je žádoucí, aby ovládací prvky na výsledném produktu byly výškově zarovnané. Zodpovědnosti byly rozděleny mezi jednotlivé moduly následovně:

- Fader Module – řízení motorizovaných potenciometrů a indikátorů signálu jednotlivých stop
- Line Module – snímání ovládacích prvků stop, ovládání displejů
- Control Module – obsluha ovládacích prvků pro kontrolu nad přehráváním a posun v nahrávce

- Automation Module – ovládací prvky pro nastavení kontroléru, výběr vlastností ovládané rotačními enkodéry Line Modulu a prvky pro ovládání automatizace a nástrojů DAW
- Master Module – hlavní modul, který řídí ostatní a tvoří abstraktní vrstvu, která není závislá na způsobu připojení k DAW

Propojení jednotlivých modulů bude řešeno sběrnici CAN bus. Tato sběrnice je používána především v automobilovém průmyslu, ale i ve sféře průmyslové automatizace. Jedná se o sériovou asynchronní sběrnici s rychlostí 1 Mb/s. Její hlavní předností je bezztrátová arbitráž, která zajišťuje, že při konfliktu na sběrnici vysílá ten uzel, který má vyšší prioritu. Díky tomu lze sběrnici použít pro přenos dat v reálném čase a garantovat deterministický přístup.



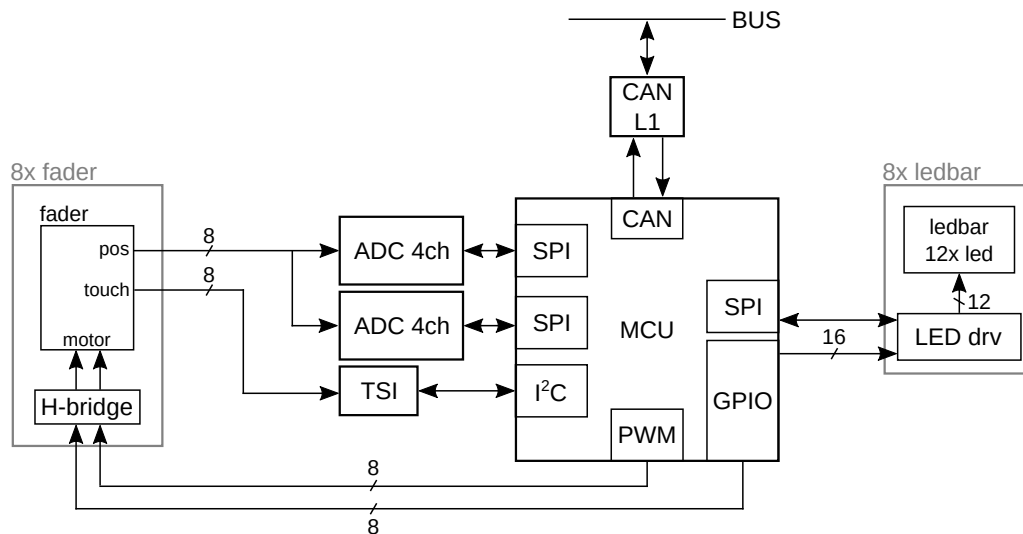
Obrázek 4.5: Blokové schéma propojení modulů. Na diagramu je znázorněn navrhovaný kontrolér (*Controller*) a volitelné rozšiřující zařízení (*Extension Unit*), se kterým bylo při návrhu počítáno

Z kapitoly 2 vyplývá, že možnost rozšíření kontroléru o další fyzické stopy je žádoucí ze strany potenciálního uživatele. Z toho důvodu byla do navrhovaného zařízení přidána podpora pro připojení rozšiřující jednotky s osmi fyzickými stopami. Tato jednotka by mohla znovu použít dva moduly navržené pro tento kontrolér: Fader a Line moduly. Ty by pak byly připojeny k hlavní sběrnici, pouze by dostaly jinou adresu, aby je bylo možné rozlišit.

Fader module

Hlavní zodpovědností tohoto modulu je řídit 8 motorizovaných potenciometrů. Jeho další funkcí je obsluhovat sloupec s LED diodami indikujícími úroveň signálu na dané stopě. Jedná se o jeden z časově nejkritičtějších modulů v systému, který musí zajišťovat precizní řízení připojených periferií v reálném čase.

Srdcem modulu bude ARM mikrokontrolér, který bude připojen na hlavní sběrnici přes CAN transceiver. Touto sběrnici bude komunikovat s Master Modulem, kterému bude posílat naměřené hodnoty z potenciometrů a přijímat instrukce k nastavení potenciometru do určité polohy nebo rozsvícení sloupce LED diod podle úrovně signálu.



Obrázek 4.6: Detailní blokové schéma Fader modulu

Hlavním funkčním celkem je řízení osmi motorizovaných potenciometrů. Potenciometry budou zapojeny jako napěťové děliče, poloha jezdce na dráze bude měřena dvěma AD převodníky. Ty budou pracovat paralelně pro dosažení požadované rychlosti převodu. Každý z nich bude připojen vlastní sběrnici SPI. Dotyk jezdce potenciometru bude snímán integrovaným obvodem dotykového senzoru. K řízení motoru bude využito periferie časovače mikrokontroléru, který bude generovat signál PWM. Spolu s logickým signálem z periferie GPIO budou řídit směr a rychlost otáčení motoru. Ten bude připojen přes H-můstek, který bude spínat potřebné napětí pro činnost motoru.

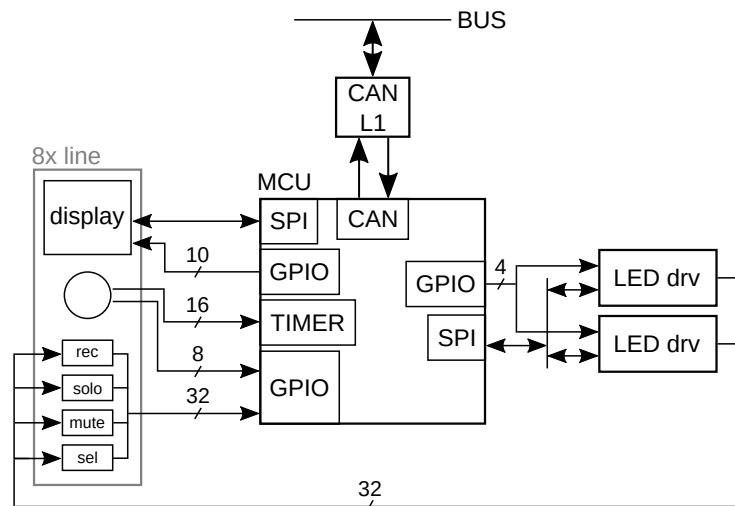
Dalším funkčním celkem je obsluha sloupců LED diod indikace úrovně signálu. Každý sloupec se skládá z 12-ti LED diod: 8x zelená, 3x oranžová a 1x červená a bude řízen jedním integrovaným obvodem pro buzení LED diod. Všechny budiče budou připojeny sběrnici SPI s využitím dvou vodičů k adresování a povolovací funkci pro každý obvod zvlášť. Budič bude mít navíc integrovanou funkci PWM pro výstupy, takže bude možné ovládat jas LED diod.

Celý modul bude realizován jako jedna DPS umístěná těsně pod motorizovanými potenciometry. Díky tomu bude možné zkrátit výkonové vodiče vedoucí k motorům na minimum. Tvar DPS bude nutné vykrojením přizpůsobit vystouplým motorům v dolní části, ve které jsou umístěny sloupce s LED diodami.

Line module

Line modul se podobně jako předchozí modul skládá z osmi stop, přičemž každá obsahuje OLED displej, rotační enkodér a 4 podsvícená tlačítka. Jeho zodpovědností je řídit tyto připojené periferie a komunikovat s Master modulem.

Ústředním prvkem je mikrokontrolér připojený k hlavní sběrnici, skrz kterou může komunikovat s ostatními moduly. Všechny displeje jsou připojeny na jedné sběrnici SPI s využitím GPIO signálů k adresování jednotlivých displejů a povolovacím funkcím společným pro všechny. Ke snímání rotačního enkodéru je použit modul časovače. Enkodér má dva výstupy, na kterých je při otáčení generovaná stejná funkce, ovšem fázově posunutá. Celkem tedy bude potřeba 16 kanálů časovače pro všechny enkodéry. Každý enkodér obsahuje na-



Obrázek 4.7: Detailní blokové schéma Line modulu

víc tlačítko, které bude snímáno spolu s ostatními tlačítky periferií GPIO mikrokontroléru. Podobně jako u předchozího modulu budou použity integrované obvody pro buzení LED diod podsvícených tlačítek, připojené přes rozhraní SPI.

Tento modul se bude skládat z jedné DPS, na které budou umístěné všechny součásti. Tím je dána i velikost plošného spoje.

Automation module

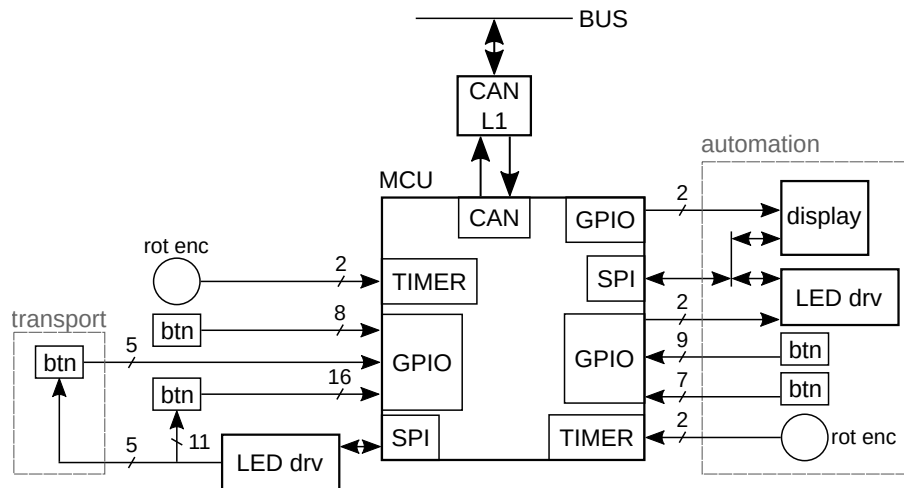
Hlavním cílem tohoto modulu je zprostředkovat uživateli přístup k automatizačním ovládacím prvkům potřebným pro práci s digitální audio stanicí a poskytnout rozhraní pro konfiguraci zařízení kontroléru. Modul byl navržen ve dvou variantách: základní a rozšířená varianta s dotykovým LCD displejem.

Základní varianta obsahuje OLED displej a rotační enkodér pro výběr vlastnosti upravené enkodéry v sekci stop a konfiguraci kontroléru a 16 podsvícených tlačítek. Z důvodu malého počtu ovládacích prvků bylo rozhodnuto vytvořit z této varianty rozšiřující modul ke Control modulu. Tento modul tedy bude obsahovat pouze popsané ovládací prvky, jejichž ovládání bude zajišťovat hostitelský modul.

Rozšířená varianta počítá s větším dotykovým LCD displejem, který bude vyplňovat celý prostor, který je k dispozici. Ten kromě funkcionality poskytované základní variantou může poskytnout mnohem pestřejší škálu možností a příjemnější uživatelské rozhraní. Tato varianta již vystupuje jako samostatný modul, který se připojí na hlavní sběrnici a bude komunikovat s ostatními moduly. Při návrhu byly učiněny kroky, aby bylo možné v budoucnu tento modul vyrobit a vytvořené zařízení jím vylepšit. V praxi se pak bude jednat hlavně o připojení ke sběrnici a napájení, společně se změnou firmwaru hlavního modulu.

Control module

Tento modul má na starost ovládání připojených periférií: 29 tlačítek (z toho 16 podsvícených) a rotační enkodér. Pokud připočteme ovládání základní varianty Automation modulu, počet periférií vzroste na 45 tlačítek (z toho 32 podsvícených), dva rotační enkodéry a malý OLED displej.



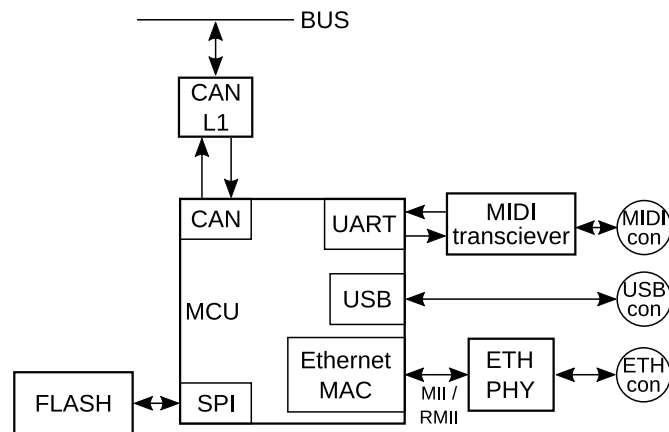
Obrázek 4.8: Detailní blokové schéma Control modulu

Základem modulu je ARM mikrokontrolér vybavený potřebnými perifériemi. Ten bude připojen k hlavní sběrnici, která mu bude umožňovat komunikaci s ostatními moduly. Z periférií budou připojeny dva rotační enkodéry (pro které je třeba vyhradit 4 kanály časovače), OLED displej a 45 tlačítek, z nichž některá budou podsvícená. Podsvícení bude řízeno stejnými integrovanými obvody jako u tlačítek modulů popisovaných výše. Z úsporných důvodů bude každý z nich na jiné SPI sběrnici, minimalizace počtu vodičů směrem k Automation modulu v základní verzi.

Modul budou tvořit 2–3 DPS (v závislosti na připojení základní varianty Automation modulu). Hlavní DPS bude zabírat většinu prostoru modulu. Její součástí bude mikrokontrolér, tlačítka a rotační enkodér. Tlačítka pro ovládání přehrávání nahrávky (na obrázku 4.8 označena jako *transport*), budou umístěna na separátní DPS z důvodu jejich jiné montážní výšky. Třetí DPS bude tvořena ovládacími prvky základní varianty Automation modulu. Obě rozšiřující DPS budou připojeny plochými kabely k hlavní desce modulu.

Master module

Master modul tvoří mozek celého zařízení a zároveň rozhraní pro komunikaci s DAW, resp. počítačem. Jeho hlavní zodpovědností je řídit ostatní moduly a vytvářet abstrakci použitého komunikačního protokolu pro zbytek systému. Jak vyplývá z předchozích kapitol, abstrakce je žádoucí, jelikož chceme komunikovat s různými nahrávacími softwary, jež používají různé protokoly pro komunikaci s kontrolérem. Při příchodí události od audiostanice proběhne překlad na událost vnitřního protokolu a ta bude odeslána příslušnému modulu ke zpracování. Stejným způsobem bude probíhat i komunikace opačným směrem – při vzniku události v modulu, kterou je třeba odeslat nahrávacímu softwaru.



Obrázek 4.9: Detailní blokové schéma Master modulu

Hlavní součástí modulu je ARM mikrokontrolér, který disponuje všemi požadovanými rozhraními: CAN, UART, USB a Ethernet. Na mikrokontroléru poběží Real-Time operační systém. Na připojené hlavní sběrnici vystupuje jako zařízení s nejvyšší prioritou. K mikrokontroléru je připojena nevolatilní paměť typu flash pro uchování konfiguračních hodnot pro celý kontrolér.

Tento modul nabízí tři rozhraní pro komunikaci s počítačem, resp. nahrávacím softwarem. Prvním je klasické MIDI rozhraní, jak jej definuje standard, včetně DIN 5 konektorů. Toto rozhraní bude odděleno optočlenem dle schématu na obrázku 3.2 a připojeno k periférii sériového rozhraní mikrokontroléru. Druhým rozhraním je USB 2.0, což je moderní rozhraní, které nalezneme na všech zkoumaných zařízeních v kapitole 2. Pro připojení bude použit konektor USB 2.0 typu B. Posledním rozhraním bude Ethernet, který je v audiotechnice stále více používaný. Primárním rozhraním však zůstává klasické MIDI a USB.

Modul bude realizován jako jedna DPS umístěná v zadní části zařízení tak, aby se konektory nacházely na pravé straně z vrchního pohledu. Nejvíce vlevo bude umístěn napájecí konektor spolu s hlavním vypínačem zařízení. Dále budou následovat dva DIN5 konektory pro klasické MIDI rozhraní, USB 2.0 B a konektor RJ45.

4.4 Zhodnocení

V rámci této kapitoly bylo navrženo řešení středně-velkého DAW kontroléru – zařízení pro ovládání nahrávacího softwaru – včetně blokových schémat jednotlivých modulů a jejich propojení. Kapitola dále diskutovala tři rozhraní, přes které je možné uskutečnit komunikaci s počítačem, resp. DAW programem běžícím na počítači.

Z důvodu vysoké časové i finanční náročnosti výroby všech navržených modulů v této kapitole bylo po dokončení fáze návrhu architektury rozhodnuto věnovat se zařízení, které bude obsahovat pouze se dva z navržených modulů. Práce se dále bude zabývat návrhem a výrobou zařízení s moduly označenými jako *Fader* a *Line*, které jsou pro ovládání digitální audio-stanice klíčové, bez újmy na dodržení zadání práce. Tyto dva moduly budou realizovány podle původního návrhu z této kapitoly, což zaručuje kompatibilitu v případě budoucích rozšíření zařízení o další z navržených modulů. Návrh nerealizovaných modulů je zde ponechán z důvodu úplnosti práce a možnosti navázání na tuto práci v budoucnu.

Kapitola 5

Návrh hardwarové platformy

Z předchozí kapitoly máme k dispozici návrh architektury dvou modulů, které budou v této kapitole realizovány z hlediska hardwaru. Nejprve bude vybrán mikrokontrolér a ostatní klíčové periferie požadované z předchozího návrhu. Následně bude vytvořena obvodová realizace za použití vybraných součástí. Po vytvoření a ověření obvodové realizace bude přistoupeno k nákresu desek plošných spojů (DPS). Posledním krokem této kapitoly je jejich výroba a osazení. Výsledek této kapitoly je zdokumentován fotografiemi v příloze C.

5.1 Fader module

V této podkapitole budou nejprve shrnuty požadavky kladené na hardwarovou realizaci Fader modulu, poté bude přistoupeno k výběru vhodných klíčových součástí. Následně bude představena nakreslená DPS a schéma zapojení.

Jádro modulu tvoří ARM mikrokontrolér běžící na dostatečné frekvenci ke zvládnutí výpočetní zátěže. U mikrokontroléru je navíc třeba počítat s dostatečným počtem periférií pro realizaci požadované aplikace. Klíčovou součástí modulu je přesný a rychlý AD převodník pro snímání polohy jezce. Pro snímání jeho doteku bude potřeba integrovaný obvod dotekového senzoru. K ovládání LED diod ledbaru slouží LED budič – posuvný registr určený k napájení více LED diod (proudově přizpůsobený). Pro řízení motoru bude použit H-můstek, který bude za pomoci napětí z pinů procesoru spínat vyšší napětí potřebné k roztočení motorů.

Komponenta	Výrobce	Typ
Spínaný zdroj	MPS	MP1482
H-můstek	Texas Instruments	DRV8870DDA
LED driver	ST	STAP16DPSP05XTTR
AD převodník	Analog Devices	AD7175-2BRUZ
Napěťová reference	Analog Devices	ADR445BRZ
Lineární regulátor +5 V	Analog Devices	ADP7104ARDZ-5.0
Dotykový senzor	Microchip	AT42QT1010-TSHR
CAN L1	Texas Instruments	TCAN332DR
Rezonátor	Abracon	ASFL1-48.000MHZ-EK-T

Tabulka 5.1: Vybrané klíčové součásti pro realizaci Fader modulu

Pro snímání polohy byl vybrán AD převodník *Analog Devices AD7175-2BRUZ*. Jelikož se jedná o čtyřkanálové převodníky, pro pokrytí všech snímaných kanálů budou potřeba dva kusy. Každý z těchto převodníků bude připojen na dedikované sběrnici. Díky tomu bude možné vzorkování paralelizovat. Převodník je napájen napětím +5 V, k jehož výrobě bude použit lineární regulátor, z důvodu menšího rušení výstupu než je tomu v případě spínaného zdroje. Digitální část převodníku je napájena napětím +3.3 V. Ke každému převodníku se váže externí napěťová reference. Pro snímání dotyku jezdce potenciometru byl vybrán jednokanálový integrovaný obvod kapacitního senzoru.

Pro řízení dohromady 96-ti LED diod ledbarů bude použito 8 LED driverů (posuvných registrů) *STAP16DPSS05XTTR*, které budou připojeny na jedné SPI sběrnici. K ovládání motorů bude použit integrovaný obvod H-můstku *DRV8870DDA* od výrobce *Texas Instruments*. Ten umožňuje za pomoci externího rezistoru nastavit maximální proud, který bude procházet obvodem.

Hlavní požadavky na rozhraní mikrokontroléru a jeho periferie vyplývající z povahy aplikace a vybraných externích periférií:

- 2x rychlé SPI pro dva převodníky. Dle manuálu převodníku [2] je minimální frekvence pro komunikaci 40 MHz.
- 1x SPI pro sběrnici budičů (posuvných registrů) ledbarů
- 8 PWM kanálů pro řízení rychlosti otáček motorů osmi potenciometrů
- 2 PWM kanály pro řízení intenzity svitu ledbarů
- 1x CAN periferie přímo na čipu mikrokontroléru pro zachování kompatibility s původním návrhem
- 8 GPIO pinů pro řízení směru otáčení motorů potenciometrů
- 8 GPIO pinů pro digitální vstup z dotykových senzorů
- 10 GPIO piny pro povolovací výstupy budičů ledbarů a AD převodníků
- 8 GPIO piny pro LED diody indikující stav + vstup stavu LDO a chybový signál převodníků
- 2x UART pro ladící výstup a interní komunikaci dvou modulů

Na základě těchto požadavků byl vybrán mikroprocesor *NXP MKV56F512VLL24*. Tabulka 5.2 porovnává požadavky spolu s vlastnostmi vybraného mikroprocesoru. Při výběru mikrokontroléru bylo vybíráno pouze z dostupných variant od výrobce *NXP*, s jehož produkty má autor zkušenosti. K vybranému MCU bude třeba externí rezonátor na frekvenci 48 MHz pro dosažení maximální frekvence jádra 240 MHz.

Z hlediska napájení bude vstupem +12 V, které bude regulováno spínaným zdrojem na +3.3 V větev. Ta bude napájet většinu integrovaných obvodů a LED diody. Další napájecí větev s napětím +5 V bude regulována lineárním zdrojem pro napájení AD převodníků.

Požadavek	NXF MKV56F512VLL24
Frekvence jádra (≥ 200 MHz)	240 MHz max
3x SPI (≥ 50 MHz)	3x (60 MHz max)
1x CAN	2x
10x PWM	16x
2x UART	5x
SWD	ok
34x GPIO	> 34x
Celkem pinů > 61	74
Flash ≥ 512 kB	1 M
RAM ≥ 128 kB	256 kB

Tabulka 5.2: Přehled požadovaných vlastností a vlastností vybraného MCU

Návrh desky plošných spojů

Navržená deska plošných spojů má rozměr 245 x 65 mm a skládá se ze čtyř vrstev. Vrchní vrstva obsahuje mikrokontrolér, AD převodníky, napájecí zdroje, fyzickou vrstvu pro CAN sběrnici a veškeré konektory. Druhá vrstva obsahuje rozlitý zemnicí polygon. Ve třetí vrstvě je rozvedeno napájení. Spodní (čtvrtá) vrstva nese H-můstky, LED drivery ovládající ledbary, napěťové reference pro převodníky a dotykové senzory.

Návrh DPS byl komplikovanější z hlediska výskytu vysokofrekvenčních digitálních signálů spolu s analogovými signály pro snímání polohy a doteku jezdců. Z toho důvodu se všechny rychlejší digitální signály vyskytují v horní třetině desky spolu s mikrokontrolérem, zatímco analogové signály jsou umístěny co nejnižší u spodního okraje desky. V místě pod AD převodníky a všemi analogovými signály je navíc rozlita analogová země, která má pouze úzký krček (resp. vyvedené plošky pro 0R rezistor) v místě u 5 V LDO regulátoru. Důvodem je minimalizace rušivých proudových smyček v oblasti analogových signálů. AD převodníky jsou orientovány analogovou stranou dolů směrem k analogovým konektorům. Druhá strana převodníku se pak směřuje přímo k mikrokontroléru. Pro předejití určitých problémů s časováním byly srovnány délky vysokorychlostních vodičů sběrnic převodníků použitím meandrů. Lineární napájecí zdroj +5 V je umístěn doprostřed mezi oba převodníky, přičemž napětí je těsně před vstupem do převodníků filtrováno feritovou perličkou. Napěťové reference se nachází z druhé strany DPS pod převodníky.

Integrované obvody dotekových senzorů jsou umístěny přímo pod konektorem pro minimalizaci délky snímaného vodiče kapacitním senzorem. Od těchto senzorů pak vedou pomalé digitální signály přes celou desku až k mikrokontroléru.

Nejvyšší frekvence se vyskytuje na sběrnicích převodníků, tyto trasy jsou ovšem poměrně krátké. Nejvíce rušivá bude pravděpodobně sběrnice propojující LED drivery s mikrokontrolérem vedoucí přes celou DPS v úrovni MCU, přestože sběrnice nebude taktována na tak vysoké frekvenci, jak je tomu u převodníků. Další rušivé obvody se nachází na horní straně desky. Jedná se o H-můstky, které budou spínány PWM signálem o frekvenci v řádech desítek kHz (př. 30 kHz, abychom předešli pískání motorů ve slyšitelném spektru). Přímo nad H-můstky se pak nachází lokální kondenzátory o kapacitě 47 uF pro pokrytí nárazové spotřeby (např. roztáčejíciho se motoru). Spínané napětí H-můstky je +12 V ze vstupní svorky – toto napětí není nikde regulované, proto by na svorky mělo být připojeno vždy napájecí napětí +12 V.

Spínaný napájecí zdroj pro +3.3 V větev je umístěn na levé straně DPS směrem k hornímu okraji, aby se co nejméně přibližoval k analogovým signálům. Pod zdrojem se nachází svorka pro vstupní napětí +12 V a hned za ní vratná proudová pojistka (polyswitch). Hned za ní se nachází rychlá dioda pro pokrytí krátkých napěťových špiček. Za diodou se vyskytuje elektrolytický kondenzátor s větší kapacitou v kombinaci s nízkokapacitním keramickým pro vyfiltrování napětí před vstupem do spínaného zdroje.

Na horní straně DPS se nachází konektory pro připojení motorů (označené jako Fx_MOT), v levém horním rohu pak konektory pro připojení ke sběrnici CAN. V blízkosti mikrokontroléru se pak nachází konektor pro ladící sériovou linku (UART0) a sériová linka k propojení obou modulů (UART3). Z praktických důvodů jsou tyto signály na dvou konektorech - na pinheaderu i na miniaturním konektoru pro použití ve výsledném systému (UART3_S). Pod MCU se pak nachází standardní 9-ti pinový konektor pro programování ARM procesorů. Na spodní straně desky lze nalézt analogové konektory pro připojení potenciometrů označené jako Fx_ANIO. Pod každým analogovým konektorem se nachází ploška pro možnost filtrování napětí feritovou perličkou.

Schéma zapojení DPS spolu s obrázky osazené DPS lze nalézt v přílohách [A](#) a [C](#).

5.2 Line module

Druhý modul obsahuje z uživatelského hlediska 32 podsvícených tlačítek, 8 rotačních enkodérů s tlačítky a 8 displejů. Dále obsahuje mikrokontrolér, budiče LED diod podsvícených tlačítek, CAN transceiver, oddělovací obvod pro MIDI a spínaný zdroj. Při jeho návrhu byla snaha znovu použít co nejvíce součástek z návrhu předchozího modulu, které se zdály být v daném zapojení funkční. Znovu použito bylo zapojení budiče LED diod pro podsvícení tlačítek, CAN transceiver a spínaný zdroj.

Pro komunikaci mezi moduly je možné využít CAN sběrnici, kterou mají všechny moduly dle původního návrhu. Pro zjednodušení byla v rámci této práce realizována interní komunikace za použití sériové linky, pro kterou byl na obou modulech vyveden SMD konektor. Oproti původnímu návrhu, kde komunikaci s počítačem zastřešoval jiný modul, bude s počítačem komunikovat tento modul s využitím MIDI komunikace, tak jak ji definuje standard. Z hlediska výpočetní náročnosti ani použitého rozhraní se nejedná o velkou zátěž navíc. Jde o sériovou linku s nízkou přenosovou rychlostí s přidanými ošetřeními vstupu a výstupu. MIDI vstup je dle standardu opatřen optoizolátorem. Jelikož specifikací doporučený typ se již nevyrobí, byl použit optočlen SHARP PC400, který dosahuje podobných vlastností, navíc v SMD pouzdře. MIDI výstup je proudově oddělen od pinu mikrokontroléru tranzistorem, aby v případě zkratu nemohlo dojít ke zničení mikrokontroléru. Ke spínání byl použit unipolární tranzistor ve variantě P-kanálu, který uzavírá obvod jeho připojením k zemi. Tím se dosáhne rozsvícení diody optočlenu v přijímacím zařízení.

Vybraný displej Visionox M01068 je připojen pomocí flexi-pásku 15-ti pinovým konektorem FFC/FPC. Jedná se o monochromatický OLED displej s řadičem SH1106. Ten obsahuje interní nábojovou pumpu k výrobě potřebného napětí pro světelné zdroje z napájecího napětí +3.3 V. Díky tomu je možné mít na DPS pouze jeden napájecí zdroj.

Displeje budou připojeny k mikroprocesoru dvěma sběrnicemi SPI pro zmenšení vyzařovaného rušení. Sběrnici bude také možné provozovat na vyšší frekvenci. Ke třetí SPI sběrnici budou připojeny integrované obvody LED driverů. Dalšími požadavky na periferie mikrokontroléru jsou 3 rozhraní UART: jedno pro konzoli ladícího výpisu, druhé pro vnitřní komunikaci mezi moduly a třetí pro MIDI rozhraní. Důležitým požadavkem na MCU je dostatečný počet kanálů časovače, které budou snímat signál z rotačních enkodérů.

Komponenta	Výrobce	Typ
Spínaný zdroj	MPS	MP1482
LED driver	ST	STAP16DPPS05XTTR
Optoizolátor	Sharp	PC400J00000F
Tranzistor P-kanál	ROHM Semiconductor	RSC002P03T316
CAN transceiver	Texas Instruments	TCAN332DR
Rezonátor	Abracon	ASFL1-48.000MHZ-EK-T
Tlačítko	Apem	Multimec 5G
Rotační enkodér	ALPS	EC11E1834403
Displej	Visionox	M01068

Tabulka 5.3: Vybrané klíčové součásti Line modulu

Celkem je potřeba 16 kanálů. Posledním požadavkem je dostatečný počet GPIO vstupních pinů pro detekci stisku tlačítek.

Na základě sepsaných požadavků byl vybrán mikroprocesor NXP MKV56F1M0VLQ24. Jedná se o stejný typ jako v případě fader modulu, ovšem ve větším pouzdře s více piny. Díky tomu bude možné později při tvorbě firmwaru sdílet kód pro obsluhu periférií mezi moduly. Tabulka 5.4 porovnává požadavky spolu s vlastnostmi vybraného mikroprocesoru. Z hlediska zapojení bude opět použit již vybraný rezonátor pro dosažení maximální frekvence jádra.

Požadavek	NXP MKV56F1M0VLQ24
Frekvence jádra (≥ 200 MHz)	240 MHz max
3x SPI	3x
1x CAN	2x
16x kanál časovače	20x
3x UART	5x
SWD	ok
59x GPIO	> 59x
pinů ≥ 94	111
Flash ≥ 1 MB	1 MB
RAM ≥ 256 kB	256 kB

Tabulka 5.4: Přehled požadovaných vlastností a vlastností vybraného MCU

Nakreslená deska plošných spojů má rozměry 240 x 96 mm. Díky nižší hustotě signálů, větším rozměrům a výskytu pouze digitálních signálů bez kombinace s analogovými bylo možné desku nakreslit na dvou vrstvách, což značně snižuje její výrobní cenu. Na vrchní vrstvě se nachází prvky uživatelského rozhraní – tlačítka, rotační enkodéry a displeje. Na spodní straně desky jsou pak umístěny veškeré ostatní součásti včetně konektorů.

FFC/FPC konektory pro displeje se nachází na horní straně desky. Propojení displejů je rozděleno na dvě sběrnice, přičemž každá propojuje 4 displeje s mikrokontrolérem a rozkládá se přes jednu polovinu DPS. Těsně pod konektory displejů se nachází rotační enkodéry, které jsou jedinou součástí vyžadující montáž skrz desku z důvodu větší mechanické odolnosti. Pod rotačními enkodéry se rozkládá 32 podsvícených tlačítek. Tato tlačítka mají

na horní straně desky připravené plošky pro případné osazení pull-up rezistorů. Ty však nebyly nakonec využity z důvodu použití interních pull-up rezistorů v mikrokontroléru.

U dolního okraje spodní vrstvy se nachází spínaný zdroj, který vytváří napětí +3.3 V, na kterém běží všechny součásti. U této desky se jeví jako výhodné neumístovat jej do rohu kvůli jednoduššímu vedení cest s napájením. Tato deska má celkem 4 svorky pro napětí. Dvě svorky slouží jako vstup napětí +12 V, dvě jsou zamýšlené jako výstup pro druhý modul. Ochrana vratnou pojistkou a rychlou diodou je stejná jako na předchozí DPS.

Zhruba uprostřed nakreslené DPS se nachází mikrokontrolér. Jeho umístění bylo zvoleno s ohledem na minimalizaci délky tras ve všech směrech. Vedle něj se nachází dva posuvné registry LED driverů zapojené za sebe – z pohledu softwaru se budou chovat jako jeden posuvný registr dvojnásobné délky. Okolo MCU jsou pak umístěny konektory pro sériovou linku výpisu do ladicí konzole (UART0) a sériovou linku vnitřního spojení s druhým modulem (UART4), jejíž signály jsou opět vyvedeny i druhým konektorem UART4_S.

Konektory pro MIDI vstup/výstup včetně potřebných oddělovacích obvodů se nachází na horní straně spodní vrstvy desky. V navrženém schématu, resp. nakreslené DPS se nachází chyba v zapojení tranzistoru pro posílení MIDI výstupu. Tato chyba znemožňovala použití výstupu, který byl držen trvale v logické 1, jelikož tranzistor se v tomto zapojení nemohl otevřít. Chyba byla provizorně opravena na již hotové DSP vhodným natočením tranzistoru mezi dvě plošky. Jeho třetí vývod byl přiveden krátkým lakovaným drátem k napájecímu napětí.

Schéma zapojení spolu s obrázky osazené DPS lze nalézt v přílohách **B** a **C**.

5.3 Testovací deska

Před realizací obou DPS navržených modulů byla vytvořena malá deska plošných spojů obsahující kritické komponenty pro tuto práci. Cílem této desky bylo odladit hardwarové zapojení ještě před návrhem obou modulů a získat tak dostatek času pro případné úpravy.

Nejkritičtější součástí bylo otestování AD převodník včetně všech podpůrných obvodů. Deska dále obsahuje konektor pro připojení vybraného typu displeje ve výrobcem doporučeném zapojení. Pro otestování ledbaru byla deska osazena 12-ti LED diodami řízenými totožným LED driverem, který byl vybrán výše. Posledními komponentami na desce jsou tlačítka a rotační enkodér, na nichž byla vyzkoušena výšková kompatibilita vzhledem k chystanému panelu.

Velikost testovací desky je 70 x 90 mm a je tvořena čtyřmi vrstvami pro odladění zapojení AD převodníku. V levém horním rohu se nachází dva pinheadery – jeden pro připojení signálů AD převodníku, druhý pro zbytek signálů. Vlevo dole se nachází analogové pinheadery pro připojení potenciometru. Všechny pinheadery mají ve vrstvě potisku popsány jednotlivé signály. Z vrchní strany je dále umístěn konektor s displejem, ledbar, rotační enkodér a dvě tlačítka. Ze spodní strany se pak nachází AD převodník včetně napěťové reference, LED driver a H-můstek.

Tento přípravek zároveň sloužil k vývoji firmwaru v době návrhu a výroby obou výše popsaných modulů. K přípravku byla připojena vývojová deska NXP FRDM-K66F obsahující mikrokontrolér s téměř identickými perifériemi k vybraným mikrokontrolérům. Výhodou tohoto přístupu byla možnost připojit sledované signály k osciloskopu a sledovat tak jejich průběh. Tento přístup se osvědčil hlavně při vývoji ovladače pro rychlé AD převodníky.

Fotografie osazené testovací desky lze nalézt v příloze **C**.

Kapitola 6

Implementace firmware

Cílem této kapitoly je přiblížit čtenáři klíčové a zajímavější pasáže z implementace firmwaru. Ten je napsán převážně v jazycích C a C++ s využitím nástrojů pro vývoj (SDK) od výrobce mikroprocesoru. K překladu byl využit překladový systém CMake. Kompletní zdrojové kódy s návodem pro překlad lze nalézt na přiloženém paměťovém médiu.

V následujících podkapitolách bude nejdříve popsána architektura firmwaru navrženého zařízení jako celku, poté bude rozebrána softwarová architektura jednotlivých modulů. Následně bude popsán způsob implementace komunikace, a to jak mezi moduly, tak s nahrávacím softwarem. Ze zajímavějších částí implementace bude popsán způsob přijímání MIDI zpráv, ovladač AD převodníků a ovládání displejů.

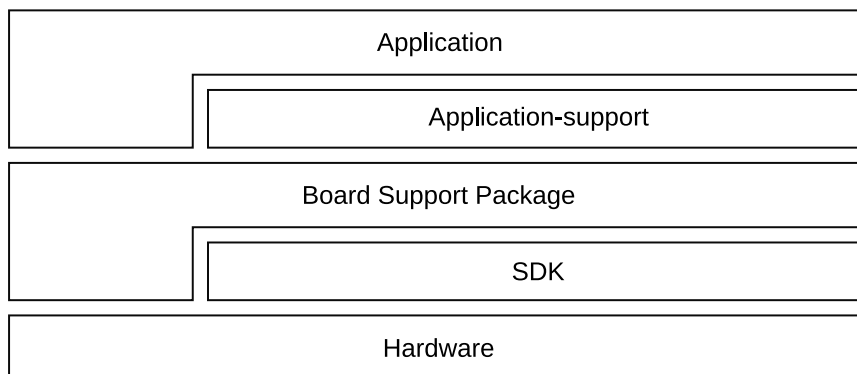
6.1 Návrh architektury firmwaru

Architektura firmwaru se odvíjí od navržené hardwarové platformy. Z předchozích kapitol máme k dispozici dvě desky plošných spojů se dvěma mikrokontroléry. Oba dva jsou stejného typu a obsahují stejné periferie. Rozdílem je velikost pouzdra a s tím spojený počet vstupně/výstupních pinů. Oba moduly se také liší ve svých požadavcích na rychlost běhu systému, latenci a v paměťových nárocích. Fader modul je konstruován pro rychlé vykonávání jedné činnosti – řízení motorizovaných potenciometrů. Line modul na druhou stranu obsahuje více různorodých úloh – větší množství ovládaných periférií, komunikace s DAW – od toho se odvíjí i větší paměťová náročnost.

Firmware se skládá z několika vrstev, rozdělení lze vidět na obrázku 6.1. Na nejnižší úrovni je samotný hardware, který je třeba ovládat. K němu máme přístup z pravidla přes registry přímo (v případě, že se jedná o periférii na čipu mikrokontroléru) nebo přes registry periferie realizující přenos dat (pokud se jedná o externí periférii mimo pouzdro mikrokontroléru). Vrstvu nad hardware tvoří nástroje pro vývoj od výrobce mikrokontroléru (SDK). Jedná se o knihovny obalující přístup k hardwaru. Obě tyto vrstvy jsou standardně poskytovány výrobcem mikrokontrolérů.

Následující vrstva je již specifická pro vytvářený projekt. Jedná se o vrstvu BSP (Board Support Package), která má odstínit vyšší vrstvy od použitého hardwaru. Vrstva BSP je společná pro oba vytvářené moduly, jelikož byl u obou použit stejný typ mikrokontroléru. V této vrstvě je většinou řešena obsluha periférií mimo pouzdro mikrokontroléru. Nad vrstvou BSP je umístěna aplikační vrstva, která byla pro přehlednost rozdělena do dvou částí. První část obsahuje podpůrné prostředky využívané aplikací daného modulu. Pro Line mo-

dul je zde například abstrakce vykreslování aplikačně specifických údajů na displej. Nejvyšší vrstvu tvoří samotná aplikace, která řídí celý modul s využitím nižších vrstev.



Obrázek 6.1: Architektura firmwaru: obrázek zachycuje rozdělení firmwaru do jednotlivých vrstev. Díky rozdělení je možné pracovat na každé vrstvě s abstrakcí předchozí vrstvy. Nejnižší dvě vrstvy jsou poskytnuty výrobcem mikrokontroléru, vrstva BSP obsahuje obsluhu základních periférií v mikroprocesoru a ovladače pro periférie mimo něj. Aplikační vrstva pak používá rozhraní z nižších vrstev a implementuje funkcionalitu.

Kvůli rozdílným požadavkům jsou na obou modulech použity různé přístupy k implementaci firmwaru. Fader modul využívá hlavní (rychlou) smyčku, zatím co Line modul používá real-time operační systém. V následujících odstavcích bude návrh firmwaru rozebrán zvlášť pro každý modul. Propojení obou modulů bude řešeno v následující podkapitole.

Architektura Fader modulu

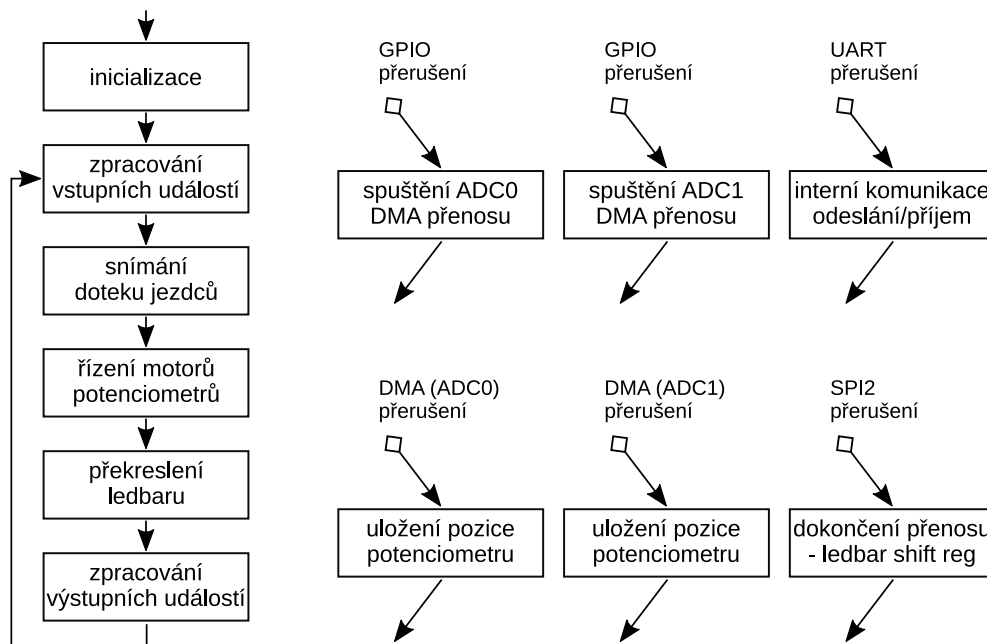
Hlavním požadavkem firmwaru pro Fader modul je dostačující rychlost pro přesné ovládání motorizovaných potenciometrů. Současně však modul musí ovládat ledbarý a komunikovat s druhým modulem. Z těchto důvodů bylo výhodné ve firmwaru použít hlavní (rychlou) smyčku, která se bude nepřetržitě starat o řízení potenciometrů. Ostatní události budou řešeny asynchronně v přerušení, resp. v něm budou ukládány do fronty, odkud budou zpracovány synchronně v hlavní smyčce. Při vhodné volbě velikosti fronty a nastavení správného objemu zpracovávaných událostí by nemělo dojít k větším výkyvům v rychlosti hlavní smyčky.

Řízení potenciometru zahrnuje snímání polohy a doteku jezdce. Dále pak ovládání motoru, který jím pohybuje. Snímání polohy jezdce je realizováno vhodným obvodovým zapojením s využitím AD převodníku. Vzorkovací frekvence převodníku musí být dostatečná ke sledování rychlých změn poloh jezdce vyvolaných motorem. Z toho důvodu jsou AD převodníky umístěny na vysokorychlostních sběrnících a v softwaru využívají DMA řadič a přerušení pro přenos dat. Signál doteku je v rámci vytvářeného firmwaru detekován jako změna logické úrovně na příslušném pinu mikroprocesoru. Jedná se o jednoduchou neblokující operaci čtení z registru periférie příslušného portu, díky tomu může být vykonávána synchronně v hlavní smyčce. Ovládání motorů je řešeno opět synchronně v každém průchodu rychlou smyčkou – jde především o neblokující nastavení PWM a GPIO periférií.

Při překreslování ledbarů probíhá komunikace s externí periférií, která trvá delší dobu. Proto bylo nutné zavést neblokující přenos. V případě detekce změny hodnoty ledbaru je přenos synchronně zahájen a poté periodicky testován s každým dalším průchodem smyč-

kou. Velikost přenášených dat do posuvného registru řídicího LED diody je 16 b, díky čemuž lze plně využít hardwarové FIFO fronty periferie řídicí sběrnici. Úspěšné odeslání je signalizováno v přerušení. V následném běhu rychlé smyčky je dokončení detekováno a v dalších průchodech jsou periodicky testovány ostatní ledbary – v každém průchodu je prováděna detekce změny právě jednoho ledbaru.

Zpracováním vstupních událostí se rozumí deserializace příchozích událostí a jejich okamžité provedení. Serializované události jsou bajt po bajtu ukládány v přerušení do fronty, odkud jsou synchronně zpracovávány. Výstupní události jsou vytvářeny při obsluze potenciometrů (změna polohy nebo dotyk) a detekovány ihned na konci smyčky. Po detekci změn jsou vytvořeny příslušné události, které se serializují do výstupní fronty a v přerušení asynchronně odešlou ke zpracování druhému modulu.



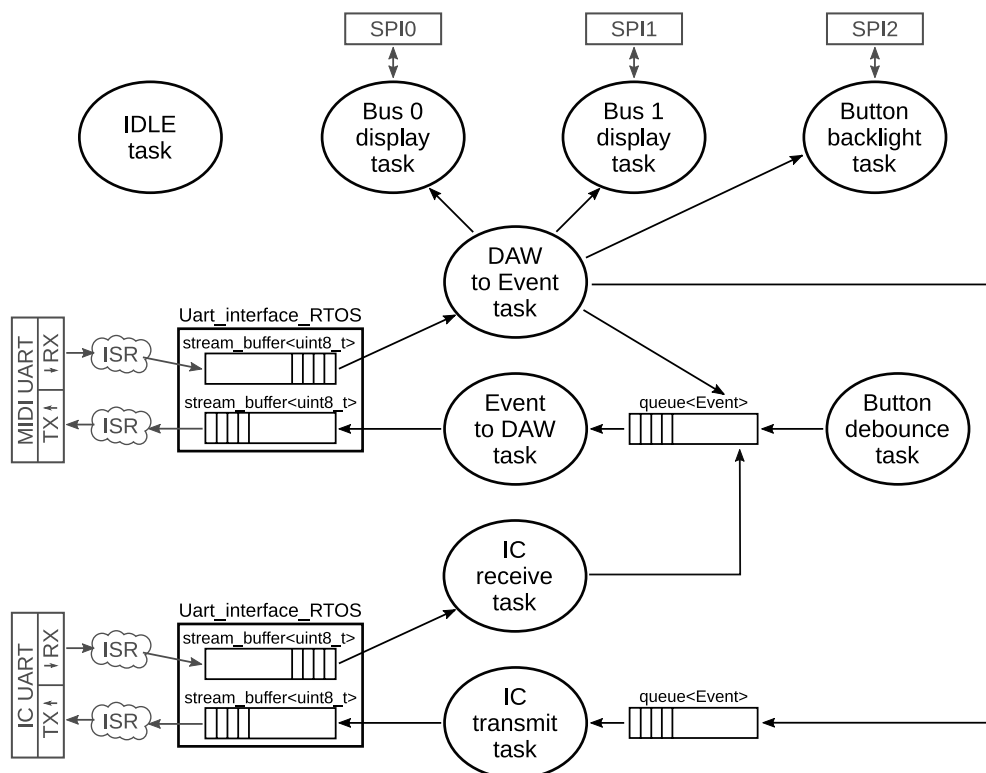
Obrázek 6.2: Hlavní řídicí smyčka firmwaru Fader modulu včetně vyznačených přerušení. Hlavní smyčka obsahuje pouze rychlé neblokující úlohy, obsluha asynchronních událostí je řešena v přerušení.

Architektura Line modulu

Požadavkem Line modulu je ovládat připojené periferie a komunikovat s nahrávacím softwarem a druhým modulem. Mezi ovládané periferie patří 40 tlačítek (z toho 32 podsvícených), 8 rotačních enkodérů a 8 displejů na dvou sběrnících. Komunikace s nahrávacím softwarem je přidaná funkcionalita pro tento modul, která nemusí být využita v případě začlenění modulu do sestavy původního návrhu z kapitoly 4.

Z důvodu většího počtu asynchronně se vyskytujících úloh i nutnosti jejich synchronizace byl na tomto modulu použit real-time operační systém *FreeRTOS*. Jednotlivé činnosti, které je třeba ve firmwaru řešit, byly rozděleny do několika úloh operačního systému.

Pro vykreslování displejů byly zavedeny dvě úlohy – jedna pro každou sběrnici, na které jsou připojeny 4 displeje. Důvodem je zvýšení rychlosti překreslování displejů paralelizací



Obrázek 6.3: Rozdělení aplikace do úloh real-time operačního systému ve firmwaru Line modulu. Na obrázku jsou vidět jednotlivé úlohy včetně způsobu komunikace a synchronizace mezi nimi. Šipka mezi dvěma úlohami představuje přímou meziúlohovou komunikaci. Pro lepší orientaci jsou jsou zobrazeny hardwarové periferie ovládané úlohami (světlejší barva).

přenosi. Z implementačního hlediska je operace probíhající na sběrnici s displejí blokující pro úlohu (nikoliv však pro procesor). Synchronizace pro překreslení displejů probíhá přímo s využitím mezi-úlohových notifikací. Notifikovaná úloha pro překreslení displeje provede nejprve vykreslení do framebufferu v paměti RAM a následně obsah této paměti přenesou do řadiče displeje. Rozlišení notifikovaných displejů je provedeno nastavením příslušného bitu v rámci notifikace. Díky tomu mohou být notifikovány různé kombinace displejů. Komunikace s displejí bude detailněji rozebrána v kapitole 6.5.

Na podobném principu pracuje i úloha řídicí posuvné registry budící LED diody podsvícení tlačítek. Oba registry jsou spojené dohromady, na sběrnici se tedy tváří jako jeden dlouhý posuvný registr. Notifikace probíhá stejně jako v předchozím případě přímou meziúlohovou synchronizací. Po obdržení notifikace je do registrů znovu nahrán aktuální obsah, příznak notifikace smazán a úloha čeká v blokováném stavu na další notifikaci. Ke komunikaci po sběrnici je využit princip přerušování – úloha je po odeslání zablokovaná a odblokovaná je opět až ve chvíli, kdy je komunikace dokončena. Blokována je pouze tato úloha, nikoliv běh celého systému.

Úloha `DAW to Event task` má za úkol přijímat komunikaci od nahrávacího programu a převést ji na vnitřní událost. Obě tyto problematiky budou detailněji rozebrány v následujících podkapitolách. Po sestavení objektu události tuto událost provede – jedná se o krátkou

neblokující činnost, kterou může být například přeposílání události do jiné fronty nebo nastavení vlastnosti objektu displeje a notifikace příslušné úloze k vykreslení.

Komunikaci směrem k počítači s DAW softwarem zastřešuje úloha `Event to DAW task`. Tato úloha setrvává v blokovaném stavu, dokud je její vstupní fronta prázdná. Po příchodu události do fronty je úloha probuzena. Následně je vyjmuta událost z fronty, převedena na zprávu, která je serializována a uložena do výstupní fronty rozhraní pro komunikaci s PC. Fronta, která synchronizuje tuto úlohu, může mít více producentů, avšak pouze jediného konzumenta (právě tuto úlohu).

Mechanická tlačítka při svém stisku generují zákmity. Proto je třeba po detekovaném stisku chvíli počkat, než se hodnota ustálí. Tato činnost je realizována v úloze `Button debounce task`, která pravidelně s určitou periodou monitoruje stav tlačítek a s pomocí modelu zjednodušeného integrátoru detekuje stisk, resp. uvolnění. Doba mezi stiskem tlačítka a detekováním stisku se liší dle mechanických vlastností tlačítka. V implementaci firmwaru byl použit čas 26 ms pro oba druhy tlačítek s rozšířením o filtr, který zabraňuje odeslání dvou stejných událostí pro stejné tlačítko za sebou. Události jsou vkládány do výstupní fronty, která je zpracovávána úlohou `Event to DAW task`.

Poslední dvě úlohy slouží ke komunikaci s Fader modulem. Obě úlohy jsou blokovány, dokud jsou jejich vstupní fronty prázdné. V momentě zaplnění fronty je událost naplánována k vykonávání. Úloha `IC receive task` přijímá serializované události od druhého modulu, deserializuje je a obslouží. Na počátku běhu zařízení (po spuštění nebo restartu) se jedná o synchronizační události, které jsou řešeny interně v rámci firmwaru zařízení, později při činnosti zařízení se jedná o události, které mají být odeslány nahrávacímu softwaru. Tyto události jsou přeposílány do vstupní fronty úlohy `Event to DAW task`, která zajistí jejich odeslání směrem k PC.

Pro odeslání události druhému modulu slouží úloha `IC transmit task`. Tato úloha je opět synchronizována vstupní frontou událostí, jež mají být odeslány. V případě neprázdné fronty tato úloha serializuje vždy jednu událost do podoby proudu bajtů, které umístí do fronty k odeslání do druhého modulu. Komunikace mezi moduly je podrobněji rozebrána v následující kapitole.

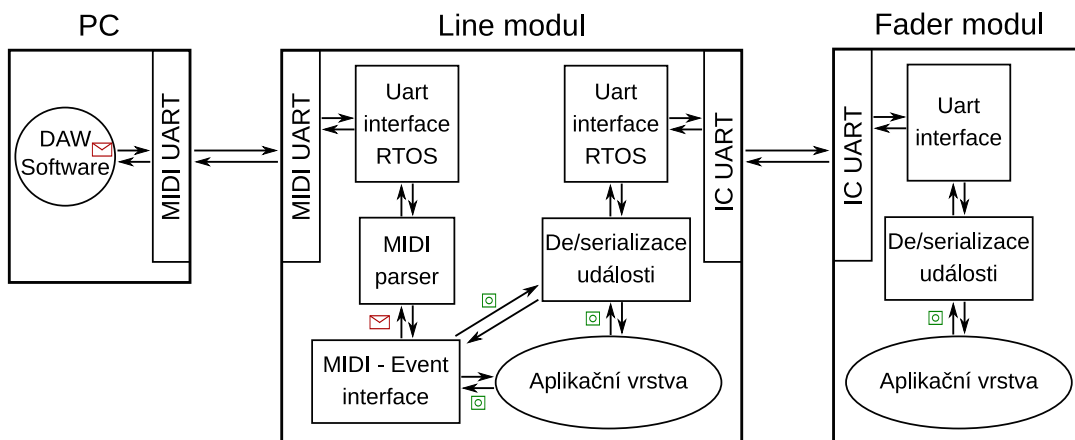
Úloha `IDLE task` má nejnižší prioritu a běží nepřetržitě – pokud není k dispozici jiná úloha s vyšší prioritou připravená k běhu. Zodpovědností této úlohy je blikání s heart-beat LED diodou, která indikuje neuváznutí modulu a udržení spojení s nahrávací stanicí pravidelným odesíláním tzv. PING zpráv.

6.2 Komunikace

Způsob komunikace se odvíjí od navržené platformy a použitého komunikačního rozhraní s nahrávací stanicí. V systému se vyskytují dva druhy komunikačních rozhraní: vnější a vnitřní. V obou případech se jedná o duplexní sériovou linku. Vnější komunikací se rozumí přenos zpráv mezi počítačem, resp. nahrávací stanicí a kontrolérem. Tento přenos je realizován v souladu s MIDI standardem na nízké rychlosti - řády desítek kb/s. Vnitřní komunikace přenáší interní události mezi dvěma moduly. V tomto případě je rychlost limitována pouze použitým hardwarem a pohybuje se v jednotkách Mb/s.

V obou případech je k implementaci využito periférií UART v kombinaci s využitím přerušení pro přijmutí, resp. odeslání bajtu dat. Logiku odesílání/přijímání zapouzdřuje třída `Uart_interface`, v případě využití operačního systému pak třída `Uart_interface_RTOS`. Obě jmenované třídy poskytují stejné rozhraní s tím rozdílem, že v případě RTOS varianty je použito synchronizace a front nabízených operačním systémem. Součástí rozhraní jsou

dvě nezávislé kruhové fronty pro vyrovnání rychlosti odesílání, resp. přijímání s rychlostí hardwarového modulu. Stejný objekt, samozřejmě s využitím jiné UART periferie, je využit i pro ladící výpis do konzole.



Obrázek 6.4: Schéma komunikace mezi moduly včetně vnější komunikace s DAW. Modelový příklad komunikace: nahrávací software posílá MIDI zprávu (červená obálka ve schématu) kontroléru. Pro odeslání ji rozloží na proud bajtů, které pošle MIDI UART rozhraním. Na straně kontroléru je přijata sekvence dat, z níž je sestavena MIDI zpráva, která je následně přeložena na vnitřní událost (zelený čtverec ve schématu) pomocí MIDI-Event rozhraní. Pokud se jedná o událost pro Line modul, je předána aplikační vrstvě. V případě události pro Fader module je tato událost serializována a přeposlána přes rozhraní IC UART příslušnému modulu. Ten provede její deserializaci a zpracuje ji.

Na úrovni aplikační vrstvy se komunikuje zasíláním událostí. Událost je nedělitelná jednotka, která nese informaci o určité akci v čase, která byla provedena nebo kterou je třeba vykonat v okamžiku výskytu. Díky použité abstrakci je aplikační vrstva zcela oddělena od komunikace s nahrávacím softwarem, která může používat různé protokoly a rozhraní. Příchozí a odchozí komunikace k počítači je převáděna na události, resp. z událostí na rozhraní vnější a vnitřní komunikace. Komunikace mezi jednotlivými moduly je řešena opět s využitím událostí, kdy je možné z pohledu aplikační vrstvy zasílat události transparentně, jako by se jednalo o zaslání události například jinému vláknu.

Na softwarové úrovni je událost tvořena objektem třídy **Event**. Tento objekt je možné serializovat, přenést bajtově orientovaným rozhraním a na druhé straně opět sestavit do objektu události se stejnými vlastnostmi. Serializovaná posloupnost bajtů může mít různou délku v závislosti na typu serializované události. Tato třída je základ, který definuje typ události a datové úložiště. Konkrétní události jsou tvořeny objekty, jejichž třídy dědí od základní třídy. Po přijetí události pak lze jednoduše přetypovat dle příslušného typu události a získat tak přístup ke specifickým atributům události.

Vnější komunikace

Vnější komunikací je komunikace nahrávacího softwaru s kontrolérem. Po sériové lince je přenášen proud bajtů serializovaných MIDI zpráv, tak jak jdou zprávy za sebou. Začátek nové zprávy vždy uvozuje Status byte. Výjimku zde tvoří Real-time zprávy, které se mohou vyskytovat kdekoli v proudu dat. Příchozí bajty na rozhraní mikrokontroléru jsou ukládány

v kruhovém bufferu instance objektu třídy `Uart_interface_RTOS`. Odtud jsou zpracovávány MIDI parserem, který z proudu bajtů opět sestaví MIDI zprávu. Přijímající automat bude popsán v podkapitole 6.3.

Tato zpráva je následně přeložena na vnitřní událost za účasti objektu třídy implementující rozhraní `IMidi_event_interface`. Jak již bylo zmíněno v předchozích kapitolách, k přenosu je ve většině případů použito MIDI rozhraní, avšak přenášené zprávy se mezi jednotlivými protokoly liší. Pokud by tedy později měla být přidána podpora více protokolů, případně jejich přepínání za chodu, stačí vyměnit instanci objektu za jinou, podporující zvolený protokol pro přenos.

Překladem MIDI zprávy na událost končí vnější komunikační rozhraní. Postup při odesílání události nahrávacímu softwaru je opačný: událost je nejprve přeložena na MIDI zprávu, která je následně serializována do podoby proudu bajtů vložených do fronty výstupního MIDI rozhraní.

Vnitřní komunikace

Vnitřní komunikace propojuje oba moduly kontroléru. Po komunikační lince jsou přenášeny serializované události. Aby bylo možné rozpoznat hranice přenášených událostí, bylo třeba zabalit přenášená data do paketu, který má definovaný začátek a konec. Z pohledu aplikační vrstvy se jedná o dvě fronty události na každém mikrokontroléru – vstupní a výstupní.

Při odesílání je první událost vyjmuta z fronty a serializována s využitím serializační metody objektu `Event`. Po získání pole datových bajtů je nutné jej zabalit do paketu přidáním oddělovačů. Díky tomu je možné spolehlivě detekovat bajtové posloupnosti jednotlivých, za sebou jdoucích, serializovaných událostí. K tomuto účelu byl využit jednoduchý protokol *SLIP* (Serial Line Internet Protocol, RFC1055[13]), který byl dříve využíván pro přenos IP datagramů přes sériovou linku. Jediným cílem protokolu je umožnit rozpoznat hranice paketů v proudu bajtů. K tomu využívá následující čtyři významově speciální bajty:

- `END` (0xC0): konec resp. začátek paketu
- `ESC` (0xDB): escape symbol
- `ESC_END` (0xDC): zakódovaný datový bajt s hodnotou `END` symbolu
- `ESC_ESC` (0xDD): zakódovaný datový bajt s hodnotou `ESC` symbolu

Kódování datové části paketu je pak přímočaré: procházíme bajt po bajtu a pokud jsou nalezeny symboly `END`, resp. `ESC`, jsou nahrazeny sekvencí `ESC-ESC_END`, resp. `ESC-ESC_ESC`. Následně je na konec paketu přidán symbol konce `END`. Dle doporučení příslušného standardu protokolu je symbol `END` přidán i na začátek paketu. Díky tomu je možné odfiltrvat veškeré chybové bajty před začátkem paketu, které může teoreticky rozhraní přijmout (např. vlivem rušení na přenosové lince).

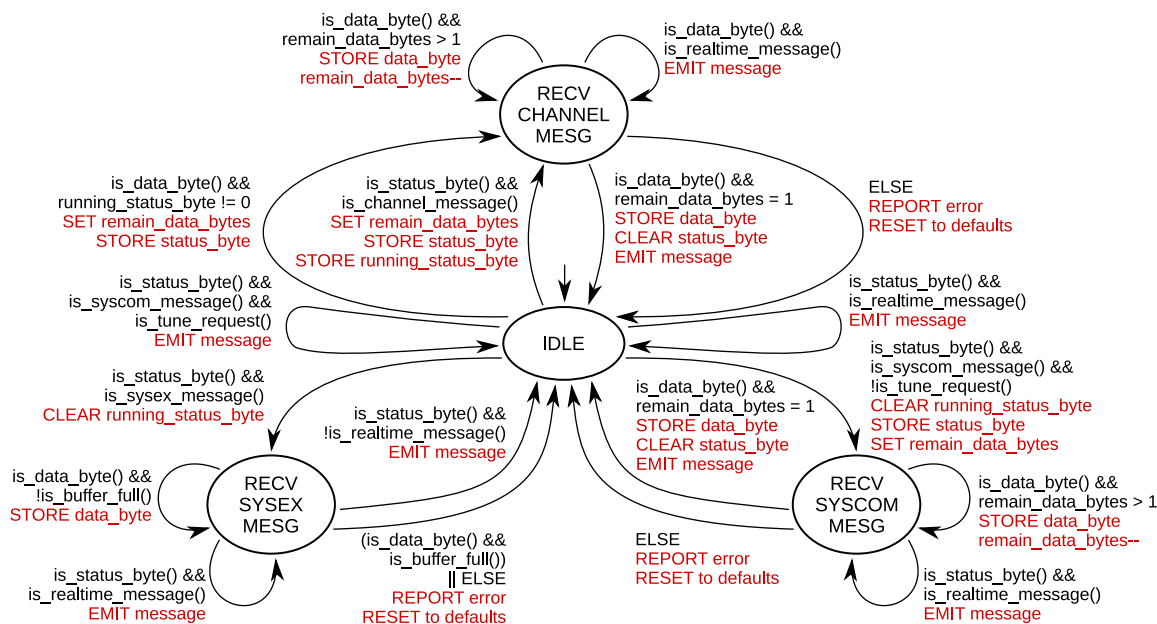
Dekódování nejprve čeká na příchozí symbol `END` signalizující začátek paketu, nevyhovující symboly před začátkem zahazuje. Poté kopíruje bajty ze vstupu na výstup. Pokud při tom narazí na `ESC` symbol, odstraní jej a následující bajt zpětně odkóduje. Při dosažení symbolu `END` je tento symbol odebrán a paket je ukončen. Následuje deserializace události metodou `deserialize` nad objektem třídy `Event`.

6.3 Přijímání MIDI zpráv

Z předchozích kapitol vyplývá, že pro přijímání zpráv máme k dispozici bajtově orientované rozhraní. Úkolem této části implementace je v příchozím proudu bajtů identifikovat jednotlivé zprávy. Svým způsobem se jedná o regulární jazyk, který může být přijímaný určitým stavovým automatem. Právě tento automat, resp. jeho zjednodušená varianta, bude popsána v této podkapitole. Zjednodušit jej bylo možné, jelikož v této aplikaci není třeba rozeznávat všechny podtypy MIDI zpráv a validovat hodnoty datových bajtů. Tento úkol je svěřen vrstvě rozhraní nad MIDI komunikací (vrstva, která převádí posloupnost MIDI zpráv na události). Hlavním cílem automatu je rozeznat hranice zpráv, jejich typ a podle něj přijmout správný počet datových bajtů. Navržený automat musí být též schopný znovu se synchronizovat v případě, kdy je na vstupu porušený bajt – například z důvodu rušení na datové lince.

Popisovaný automat je jádrem třídy `MIDI_parser`. Veřejným rozhraním této třídy je metoda `put_byte`, která jako vstup přijímá jeden bajt přijatých dat od nahrávací stanice a ukazuje na MIDI zprávu, do které bude uložen obsah v momentě přijetí kompletní zprávy automatem. Výstupem této metody je pravdivostní hodnota, zda odkazovaná zpráva na vstupu obsahuje přijatou validní MIDI zprávu.

MIDI zpráva je reprezentována třídou `MIDI_message`. Ta dovoluje pohodlně manipulovat s typem zprávy a bezpečně pracovat s datovými bajty bez rizika přístupu mimo pole datových bajtů. Velikost tohoto pole je shora omezená a přizpůsobená délce přenášených dat nejdelším typem zprávy – System Exclusive Message. Pro pohodlnější vývoj lze zprávy vypsat na obrazovku pomocí třídy `MIDI_dumper`.



Obrázek 6.5: Zjednodušený stavový automat přijímající MIDI zprávy. Podmínky pro přechod mezi stavy jsou vyznačeny na hranách černou barvou, akce provedená při přechodu červenou.

Přijímající automat se na počátku nachází ve stavu *IDLE*. V tomto stavu se čeká na příchod stavového (Status) bajtu. Veškeré datové bajty jsou ignorovány. Při přijetí stavového bajtu se dle jeho typu rozhodne, do kterého stavu se přejde. Pokud se jedná o zprávu typu *Channel Message*, automat přejde do stavu *RECV CHANNEL MESSG*, současně si uloží přijatý stavový bajt a určí počet datových bajtů k přijetí pro daný typ zprávy. Následně se přijímá nastavený počet datových bajtů. V případě přijetí všech datových bajtů je MIDI zpráva přijata (naznačeno událostí *EMIT* na obrázku 6.5). Pokud však není přijatý dostatečný počet datových bajtů, použije se druhá hrana pro přechod do stavu *IDLE*, ve které je chyba oznámena událostí *REPORT* (pouze ve vývojářském prostředí) a stavový automat přejde do výchozího stavu, přičemž obnoví veškerý vnitřní stav na výchozí hodnoty.

Přijímání zpráv typu *System Common* probíhá stejným způsobem. Důvodem k rozdělení této funkcionality do dvou stavů je podpora vlastnosti Running-status MIDI protokolu v případě přijímání zpráv typu *Channel Message*. Přijímání zprávy typu *System Exclusive Message* probíhá ve stavu *RECV SYSEX MESSG*. Zpráva tohoto typu nemá pevně danou délku datové části, z toho důvodu automat ukládá znaky, dokud není vnitřní buffer plný. Pokud tento stav nastane, opět probíhá zotavení do výchozího stavu bez přijetí zprávy. V případě přijetí speciálního bajtu ukončujícího tento typ zprávy (bajt *EOX*) nebo jiného stavového bajtu je zpráva dokončena, přijata a přechází se do výchozího stavu. Ve všech stavech lze navíc na libovolné pozici přijímat jednobajtové zprávy typu *Real-Time Message*.

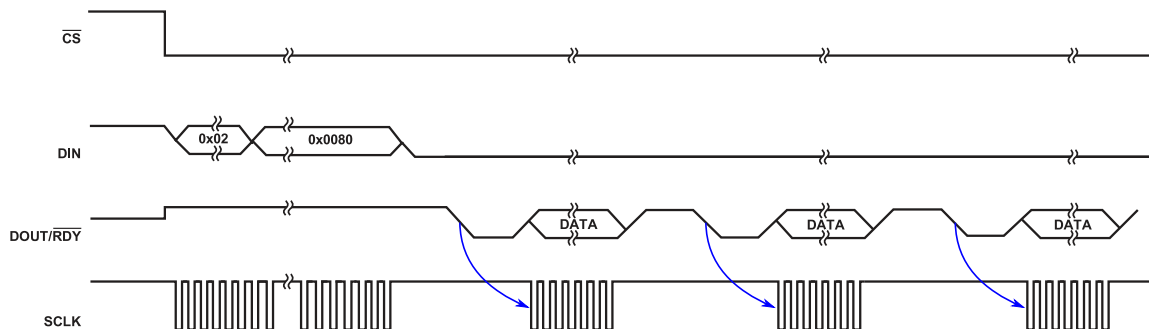
Opačným úkolem k přijetí MIDI zprávy je její odeslání směrem k počítači. V tomto případě máme k dispozici od vyšší vrstvy (vrstva rozhraní) jednu nebo i více MIDI zpráv, reprezentující jednu událost kontroléru. MIDI zpráva musí být před odesláním nejprve serializována do posloupnosti bajtů. K serializaci slouží třída *MIDI_message_serializer*. Ta obsahuje čistě virtuální metodu *send_byte*, kterou je nutné implementovat pro použití v konkrétním systému. Díky tomu je serializér nezávislý na použité nižší (transportní) vrstvě. V této implementaci metoda pouze přidává bajty do výstupní fronty *MIDI UART* rozhraní.

6.4 Ovladač AD převodníku

V kapitole 5.1 byl pro měření polohy motorizovaných potenciometrů vybrán AD převodník *Analog Devices AD7175-2*. Pro pokrytí všech kanálů kontroléru byly na DPS umístěny 2 ks těchto převodníků: každý z nich má vlastní dedikovanou sběrnici k mikrokontroléru. Použité převodníky nabízí vysokou míru konfigurovatelnosti, avšak na úkor jednoduchosti obsluhy. Implementované aplikační rozhraní pokrývá nastavení většiny registrů a podporu pouze těch módů, které jsou skutečně používány.

Nastavení převodníku probíhá v módu *Standby*, kdy je funkční pouze digitální část převodníku, analogová část je vypnutá. Pro čtení navzorkovaných hodnot z převodníku je použit *Continuous-read* mód. V tomto módu převodník automaticky cyklí nad povolenými kanály k převodu. Zároveň je také ignorována veškerá komunikace směrem k převodníku. Změna na jiný mód je možná provedením restartu digitálního rozhraní, který lze provést softwarově. V tomto módu je využit sdílený výstupní signál *DOUT/nRDY* převodníku. Po přechodu do tohoto módu mikrokontrolér monitoruje signál *nRDY* a čeká na jeho sestupnou hranu, která značí dokončení převodu. Následně mikrokontrolér spustí čtení a přečte navzorkovanou hodnotu.

V navrženém kontroléru byly využity všechny čtyři nabízené kanály, všechny sdílejí stejné nastavení. Pro snížení objemu přenášených dat je do mikrokontroléru přenášen pouze 16-bitová hodnota navzorkovaného signálu. V převodníku je navíc zapnutá volba, která



Obrázek 6.6: Diagram průběhu signálů na převodníku během *Continuous-read módu*. V prvním čtecím/zápisovém cyklu lze vidět přechod do *Continuous-read módu*. V následných čtecích cyklech sestupná hrana signálu *DOUT/nRDY* indikuje dokončení vzorkování a způsobí čtení vzorku. Během čtení registru probíhá vzorkování následujícího kanálu. Po navzorkování signálu na vstupu je připravenost ke čtení signalizována opět sestupnou hranou. Převzato z [2], upraveno.

umožňuje při použití módu přidat za navzorkovanou hodnotu obsah stavového registru. Díky tomu lze zjistit aktuální stav převodníku a kanál, jehož hodnota byla vzorkována.

Maximální vzorkovací frekvence AD převodníku je 250 000 vzorků/s. Z toho důvodu se předpokládá rychlá komunikace po sběrnici. V implementovaném systému je rychlost sběrnice nastavena na maximální, kterou periférie dovoluje – 60 MHz. Použitá vzorkovací frekvence je však nižší. Z důvodu zvýšení rychlosti a snížení zátěže procesoru byla pro přenos dat po sběrnici použita jednotka DMA v kombinaci s přerušením: V momentu sestupné hrany signálu *DOUT/nRDY* je vystaveno přerušování, ve kterém je multiplexorem přepnut pozorovaný signál do periférie SPI. Následně je zahájen přenos pomocí DMA. Po dokončení přenosu je opět v přerušování tento signál přepnut do periférie GPIO, kde se čeká na další sestupnou hranu.

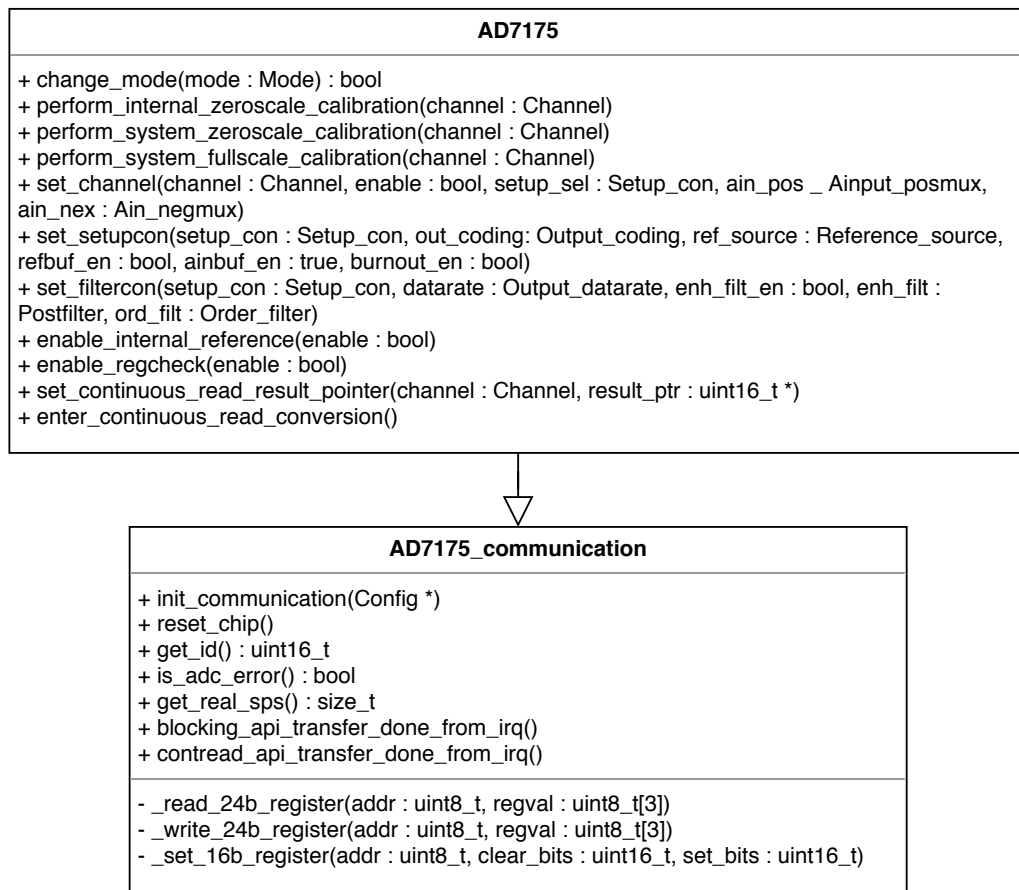
Požadavkem pro tvorbu ovladače převodníku byla rychlá komunikace po sběrnici při čtení vzorků a uživatelsky přímočaré nastavení převodníku při jeho inicializaci. Z toho důvodu bylo poskytované aplikační rozhraní rozděleno na dvě skupiny:

- Blokující API – pro nastavení vlastností převodníku při inicializaci firmwaru
- Continuous API – pro rychlé autonomní čtení převedených hodnot z převodníku

Blokující aplikační rozhraní slouží k uživatelsky přívětivému nastavování vlastností AD převodníku. Jedná se o výchozí mód, ve kterém se objekt třídy `AD7175` nachází po vytvoření. V tomto módu lze s převodníkem komunikovat pomocí metod, které nabízí jmenovaná třída. Lze spustit kalibraci celého převodníku i jednotlivých kanálů, nastavit pozitivní i negativní zdroj měřených signálů v rámci kanálu, případně další vlastnosti kanálu. Dále lze nastavit pro každý kanál zvlášť konfiguraci kódování, vzorkovací frekvence a reference. Pro každou z konfigurací lze pak dále nastavit vlastní filtr. Přehled vybraných metod lze nalézt v diagramu tříd na obrázku 6.7.

V implementovaném firmwaru je použita konfigurace, která každému kanálu přiřazuje vlastní *setupcon* registr s nastavením. Teoreticky by bylo možné všechny kanály nasměrovat na jednu konfiguraci. Přišli bychom však později o možnost využít kalibraci převodníku pro každý kanál zvlášť, jelikož kalibrační data jsou vázána právě na tento registr.

Continuous aplikační rozhraní tvoří metoda `set_continuous_read_result_pointer`, která pro specifikovaný kanál uloží ukazatel na paměťové místo, kam bude ukládána aktuální navzorkovaná hodnota signálu na tomto kanálu. Po nastavení všech ukazatelů se přejde do *Continuous-read* módu voláním metody `enter_continuous_read_conversion`. Od této chvíle již nelze používat blokuující aplikační rozhraní (volání skončí `assertem`). Těsně před spuštěním tohoto režimu v implementovaném firmwaru je zapnuta funkce přidávající status register za přijatý vzorek a funkce pro kontrolu integrity registrů s nastavením. Pokud by se tedy změnil obsah jakéhokoliv registru s nastavením, AD převodník by tuto skutečnost oznámil nastavením bitu ve stavovém registru.



Obrázek 6.7: Diagram tříd implementace ovladače AD převodníku Analog Devices AD7175-2. Implementace je rozdělena do dvou tříd. Virtuální třída `AD7175_communication` má zodpovědnost za nízkoúrovňovou komunikaci s AD převodníkem. Třída `AD7175` od ní dědí má na starost vysokoúrovňové API pro nastavení převodníku.

6.5 Ovladač displeje

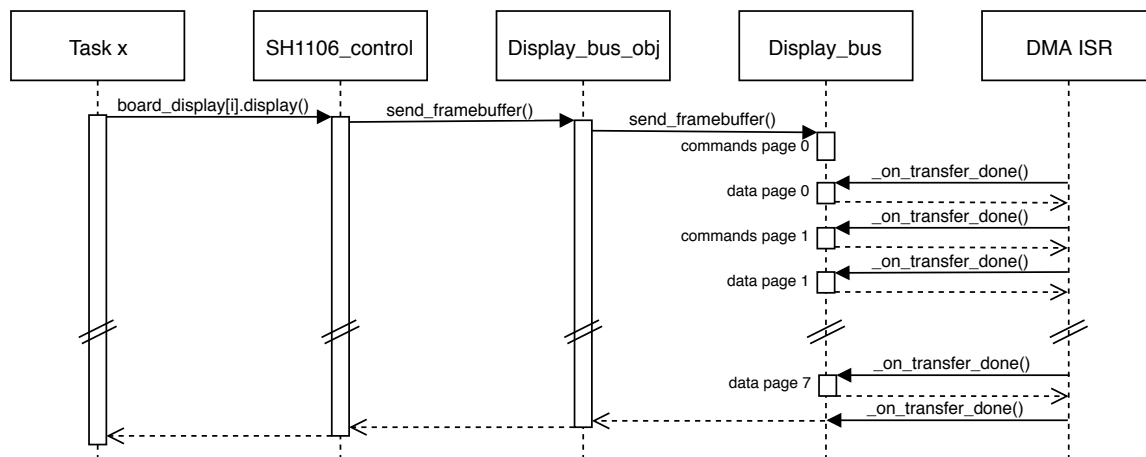
Jako zobrazovací zařízení umístěné na Line modulu byl vybrán displej *Visionox M01068*. Jedná se o monochromatický displej typu OLED s řadičem SH1106. Tento typ displeje je použit celkem osmkrát, každá polovina je připojena na svoji vlastní sběrnici.

Rychlost přenosu dat není tak kritická jako v případě AD převodníku, obě sběrnice běží na frekvenci 20 MHz. Komunikace s displejem je pouze jednostranná, rozhraní displeje ne nabízí signály pro komunikaci směrem k mikrokontroléru. Z toho důvodu je třeba uchovávat v paměti framebuffer, který bude po vykreslení přenesen do displeje. Pro přenos framebufferu do řadiče displeje je použito DMA, jelikož se jedná o větší množství dat (velikost framebufferu je 768 B), které lze odesílat téměř bez účasti procesoru.

Z hlediska softwaru je implementace rozdělena do tří tříd starajících se o vykreslování požadovaných informací. Nejnižší vrstvou je použitá knihovna třetí strany `Adafruit_GFX` (BSD licence). Tato knihovna se stará o vykreslování základních grafických primitiv. Jedná se však pouze o čistě virtuální knihovnu, která nezná hardware displeje, na který se bude vykreslovat.

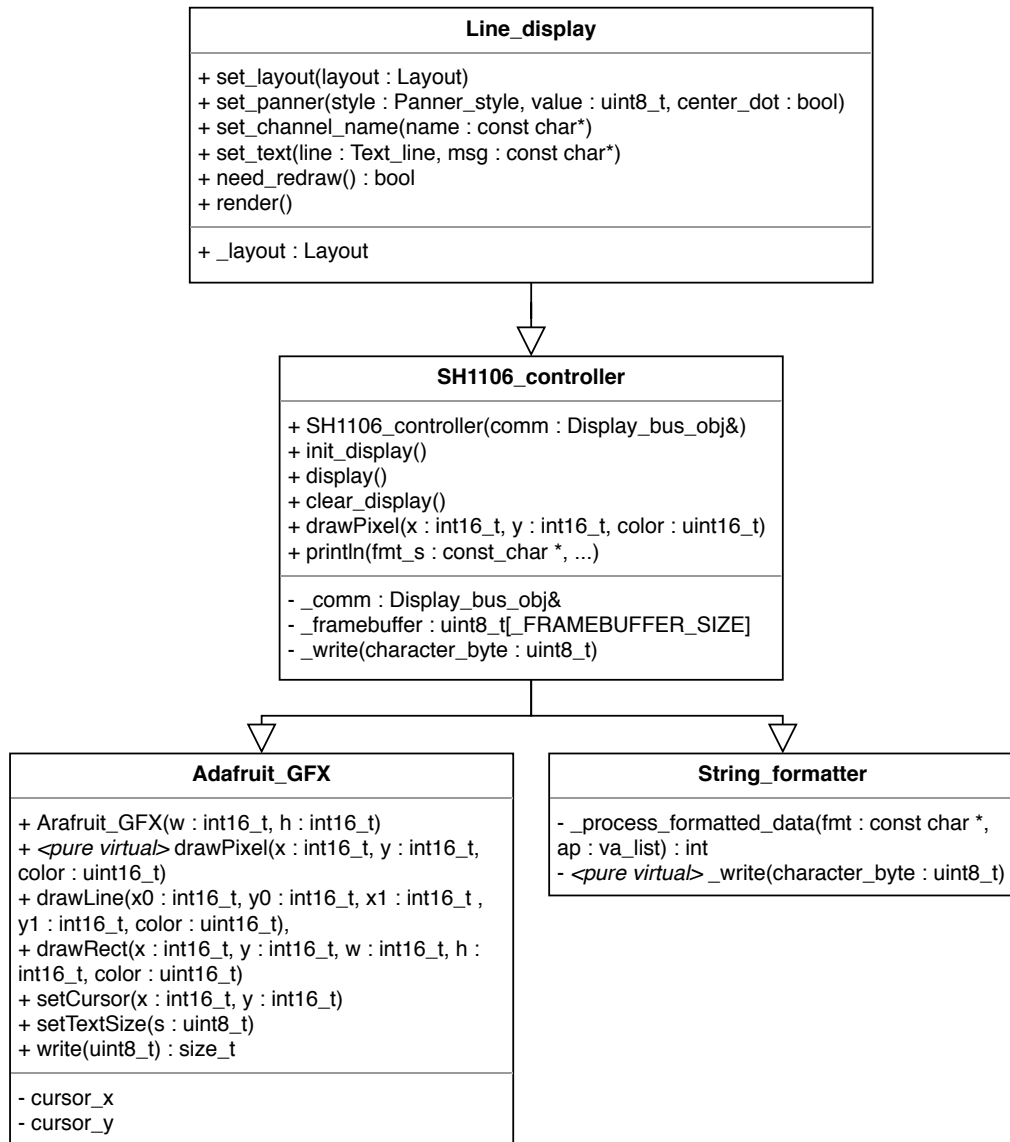
Druhou vrstvou je třída `SH1066_controller`, která byla implementována v rámci této práce. Její zodpovědností je práce s framebufferem a komunikace s displejem – odeslání vykresleného framebufferu do řadiče displeje. Tato třída dědí od třídy `Adafruit_GFX` a implementuje virtuální metodu `drawPixel`, pro kreslení pixelu do framebufferu. Přenos framebufferu do řadiče displeje je zahájen metodou `display`. Tato třída navíc používá vícenásobnou dědičnost. Jejím druhým předkem je třída `String_formatter`, která umí z formátovacího řetězce a proměnného počtu parametrů vytvořit řetězec pro zobrazení na displej. Pro zprovoznění této funkcionality bylo nutné implementovat její virtuální funkci `_write`, která pouze předá vykreslovaný znak použité grafické knihovně a ta jej vykreslí (metoda `write`). Spojením těchto tříd je možné na displej zobrazit text nebo jakýkoliv geometrický útvar.

Přenos framebufferu do řadiče displeje obstarává řadič DMA s využitím přerušení. Řadič displeje má paměť interně rozdělenou na 8 stránek, přičemž tento typ řadiče neumí automaticky inkrementovat číslo stránky, ale pouze číslo sloupce. Z toho vyplývá, že největší velikost dat, kterou lze displeji poslat najednou, je velikost stránky (96 B). Poté je nutné řadiči displeje odeslat příkaz pro změnu stránky a poslat data další stránky. Tímto způsobem lze odeslat displeji celý framebuffer.



Obrázek 6.8: Sekvenční diagram volání metod při odesílání framebufferu do řadiče displeje. Cílem implementace byla rychlost odeslání dat do řadiče v kombinaci s co nejmenší zátěží procesoru. Z toho důvodu bylo využito DMA. Displej však nepodporuje automatickou inkrementaci stránky, proto je nutné posílat data po částech.

Nejvyšší vrstvu tvoří třída `Line_display`. Ta má na starost vykreslování specifických rozložení pro aplikaci kontroléru. V rámci aplikace je použito několik různých rozložení obrazovek, které jsou nazvány *layouty*. K dispozici je například layout pro logo kontroléru zobrazovaném při startu nebo rozvržení použité při komunikaci s DAW. Rozvržení je nastaveno metodou `set_layout`, poté jsou nastaveny všechny potřebné informace k vykreslení (metody s předponou `set_`). Následně je zvolené rozvržení vykresleno do framebufferu metodou `render`.



Obrázek 6.9: Diagram tříd implementace ovladače displeje. Na diagramu je zachycena struktura dědičnosti tříd od nejvíce vysokoúrovňového API tvořeného třídou `Line_display` přes třídu `SH1106_controller` starající se o vykreslování obrázku a přenosu dat do displeje až po třídu `Adafruit_GFX` použité knihovny pro vykreslování grafických primitiv.

Kapitola 7

Sestavení a testování zařízení

Výsledkem předchozích kapitol je funkční prototyp DAW kontroléru skládající se ze dvou desek plošných spojů a osmi motorizovaných potenciometrů. V této kapitole bude popsán způsob složení těchto komponent do jednoho prototypového zařízení a způsob jeho testování v reálném prostředí nahrávacího softwaru.

7.1 Výroba panelu zařízení

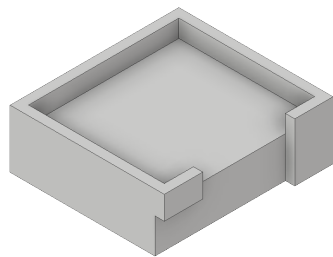
Návrh panelu vychází z levé části nákresu uvedeném na obrázku 4.4. Panel obsahuje v horní části výřezy pro displeje, rotační enkodér a podsvícená tlačítka. Ve spodní polovině panelu se nachází výřezy pro jezdcy motorizovaných potenciometrů a vedle nich v dolní části výřezy pro ledbary. Návrh tohoto panelu byl koncipován jako prototyp, z toho důvodu byly použity distanční sloupky délky 35 mm jako podpěry, aby bylo možné do něj nahlédnout ze strany desek plošných spojů.

Line modul je k panelu přichycen distančními sloupky o výšce 10 mm v osmi bodech vybraných tak, aby stisk tlačítka měl co nejmenší dopad na prohyb desky plošných spojů. Displeje jsou zvednuty až do výšky, kterou umožňuje jejich flexi-pásek pro připojení ke konektoru. Aby bylo možné displeje zafixovat ve zvednuté poloze, byly navrženy a na 3D tiskárně vytištěny podložky, které displeje obejmou ze čtyř stran a podrží je v poloze nad DPS.

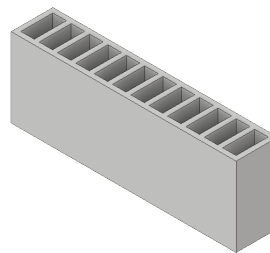
Fader modul je umístěn přímo pod motorizovanými potenciometry v místě mezi motorem a terminály pro připojení vodičů. Díky tomu mohou být vodiče z potenciometru ke konektorům na DPS relativně krátké, což se pozitivně projevilo na intenzitě šumu při měření signálu AD převodníky. Modul je zvednut nad panelem distančními sloupky o výšce 11 mm. Vedle každého potenciometru je ve spodní části panelu umístěn výřez pro zasazení ledbaru, který je součástí modulu. Pro dodržení stejných odstupů jednotných LED diod byl na 3D tiskárně vytištěn držák, který pojme všechny LED diody ledbaru. Tento držák leží přímo na DPS a je zarovnan s horní hranou panelu.

Navržené rozložení bylo vyfrézováno do 2 mm silného hliníkového panelu, který byl následně povrchově upraven s využitím techniky eloxování hliníku na černou barvu. Upravený panel byl potištěn za použití technologie UV tisku.

Nákres panelu je umístěn v příloze D.



(a) Model podložky displeje



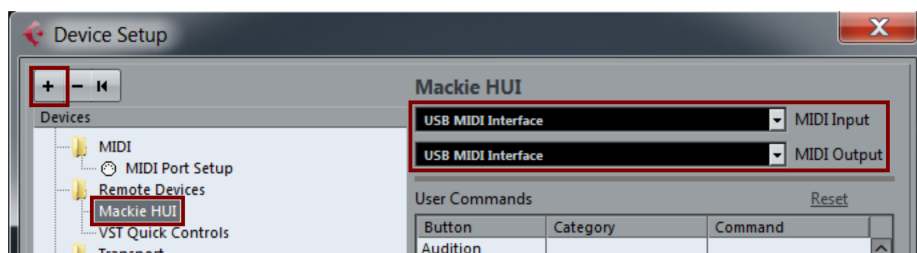
(b) Model držáku LED diod ledbaru

Obrázek 7.1: 3D modely součástí tisknutých na 3D tiskárně

7.2 Testování zařízení

Testování prototypu probíhalo v zapojení s reálným nahrávacím softwarem *Steinberg Cubase Elements 8*. Propojení s počítačem je realizováno USB-MIDI převodníkem, který je zapojen jednou stranou do USB portu počítače a druhou stranou do vytvořeného DAW kontroléru.

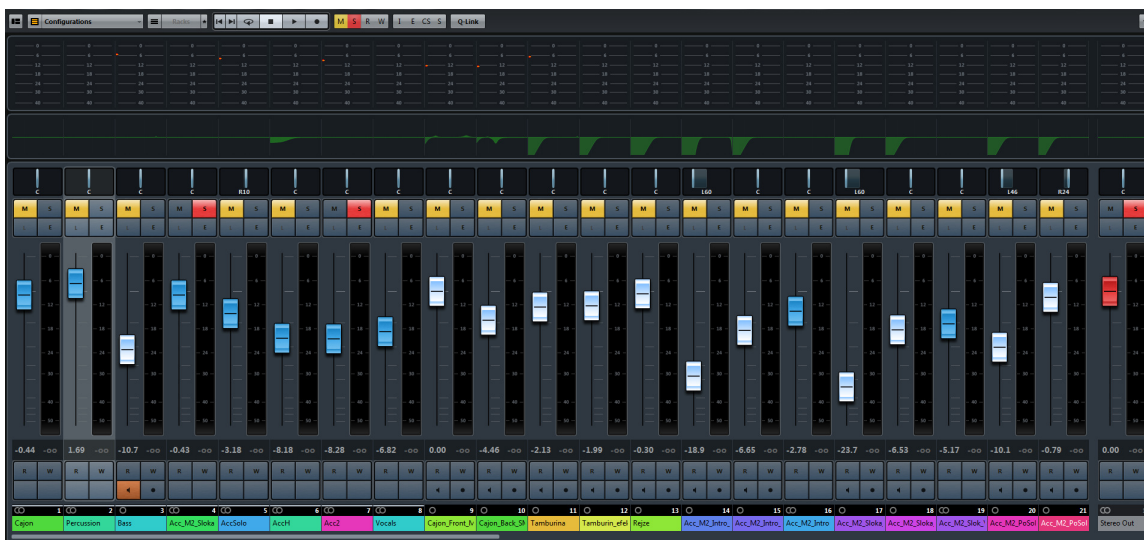
Nahrávací software je třeba před připojením kontroléru nastavit. Nastavení spočívá v přidání kontroléru a vybrání MIDI portů, které budou použity pro komunikaci. Po příslušném nastavení rozhraní je možné kontrolér připojit a zapnout.



Obrázek 7.2: Nastavení připojení kontroléru v nahrávacím softwaru *Steinberg Cubase*

V rámci testování byl v DAW softwaru vytvořen projekt s osmi zvukovými stopami, abychom pokryli všechny jezdce kontroléru. Testovány byly jednotlivé součásti uživatelského rozhraní:

- Displej – zobrazuje název stopy a ukazatel natočení enkodéru. Při změně jedné z hodnot se překreslí.
- Tlačítka – reagují na stisk, podsvícení se shoduje s virtuálním mixem v programu. Mapování tlačítek je korektní. Nenastane případ, kdy se při stisku fyzického tlačítka provedou dva stisky v DAW mixu.
- Ledbary – ukazují aktuální sílu signálu při přehrávání, která vizuálně koresponduje s hodnotou ve virtuálním mixu. Při zastavení přehrávání postupně zhasnou.
- Jezdce – Při posunu jezdce není cítit odpor motoru. Současně s jezdcem se pohybuje i zeslabovač ve virtuálním mixu. Funguje i opačný směr: při pohybu zeslabovače ve virtuálním mixu se současně pohybuje jezdec na kontroléru. Vyzkoušení funkce doteku provedeno zapnutím nahrávání automatizace – dotek odstartuje nahrávání.



Obrázek 7.3: Snímek obrazovky projektu v programu *Steinberg Cubase*



Obrázek 7.4: Testovaný kontrolér připojený k digitální audiostanici z předchozího obrázku

Kapitola 8

Závěr

V rámci této práce byl navržen a realizován kontrolér pro ovládání digitální audio stanice. Jedná se o vestavěné zařízení se dvěma mikrokontroléry, jehož uživatelské rozhraní je tvořeno osmi motorizovanými potenciometry, ledbary, tlačítky, rotačními enkodéry a displeji. Cílem zařízení je zvýšit komfort audio inženýra a zrychlit jeho pracovní proces.

První část práce se věnuje termínu digitální audio stanice a zasazení vytvářeného zařízení do kontextu nahrávacího studia. Po vytvoření základní představy o zařízení jsou uvedeny tři případové studie kontrolérů, které lze nalézt na trhu. Z těchto krátkých studií je vyvozen závěr, od kterého se odvíjí navrhované řešení. Další část se zabývá problematikou komunikace s kontrolérem. V ní jsou blíže prozkoumána používaná rozhraní, kterými lze spojit kontrolér s počítačem, resp. nahrávací stanicí. Dále je rozebrán MIDI protokol a jedna z jeho existujících nadstaveb, pomocí které komunikuje digitální audiostanice s kontrolérem.

Ve třetí části práce je představen motorizovaný potenciometr, který je ústředním prvkem uživatelského rozhraní vytvářeného zařízení. Poté je navržena architektura kontroléru a systém je dekomponován do několika modulů. Následující část se zabývá hardwarovým návrhem dvou modulů dle navržené architektury a jejich realizací do podoby funkčních celků. Zde jsou navrženy, vyrobeny a oživeny dvě desky plošných spojů. V páté části je pro každý z modulů vytvořen firmware. Jelikož se jedná o řešení založené na dvou mikroprocesorech, výsledkem je binární program pro každý z nich. V této části je detailně rozebrána architektura firmwaru obou modulů, jejich propojení do jednoho celku a vytvoření aplikace ovládající periferie kontroléru. Poslední část se zabývá výrobou hliníkového panelu, do kterého jsou umístěny oba vytvořené moduly a popisuje způsob, kterým byl vytvořený kontrolér testován na reálném nahrávacím softwaru.

Výsledkem této práce je funkční prototyp DAW kontroléru, tedy vestavěného zařízení, kterým lze ovládat software digitální audiostanice. Tento prototyp byl úspěšně otestován s nahrávacím softwarem *Steinberg Cubase*. V rámci práce byla navrhována architektura zařízení, která se skládá z více modulů, než bylo realizováno. Současně s tím bylo při realizaci hardwarových modulů z důvodu možné budoucí kompatibility postupováno podle navržené architektury. Díky tomu může být vytvořené zařízení v budoucnu rozšířeno o další z navržených modulů do podoby většího kontroléru. K němu může být navíc připojen další (rozšiřující) kontrolér navyšující počet simultánně ovládaných stop digitální audiostanice. V případě přidání dalších modulů je nezbytné rozšířit obslužný firmware.

Literatura

- [1] Alps: *Motor-driven Master Type*. 2019, [Online; navštíveno 7.1.2019].
URL <https://www.alps.com/prod/info/E/PDF/Potentiometer/Slide/RSN1M.pdf>
- [2] Analog Devices: *AD7175-2 datasheet Rev. B*. 2014-2016, [Online; navštíveno 2.4.2019].
URL <https://www.analog.com/media/en/technical-documentation/data-sheets/AD7175-2.pdf>
- [3] Apple Computer, Inc.: *Logic Pro 7.2.1: Dedicated Control Surface Support*. 2006, [Online; navštíveno 11.12.2018].
URL https://people.ok.ubc.ca/creative/MacManuals/Logic7_DedicatedCntrlSurfaceInfo.pdf
- [4] Ashour, G.; Brackenridge, B.; Tirosh, O.; aj.: *Universal Serial Bus Device Class Definition for MIDI Devices*. Listopad 1996, [Online; navštíveno 10.1.2019].
URL <https://www.usb.org/sites/default/files/midi10.pdf>
- [5] Conrad Electronic: *Motorizovaný tahový potenciometr ALPS RSAON11M9*. 2019, [Online; navštíveno 14.1.2019].
URL <https://www.conrad.cz/motorizovany-tahovy-potenciometr-10-kohm-0-5-w-linear-ni-alps-rsaon11m9-1-ks.k442081>
- [6] Lazzaro, J.; Wawrzyneki, J.: *RTP Payload Format for MIDI*. RFC 6295, RFC Editor, June 2011.
URL <http://www.rfc-editor.org/rfc/rfc6295.txt>
- [7] LOUD Audio, LLC: *MCU Pro Control Surface - Mackie*. 2018, [Online; navštíveno 5.1.2019].
URL <https://mackie.com/products/mcu-pro-and-xt-pro>
- [8] Michael Slinn: *Mackie HUI Protocol*. 1994-2019, [Online; navštíveno 15.3.2019].
URL <https://www.mslinn.com/sites/mike/studio/midi/MackieControlMIDIMap.html>
- [9] MIDI Manufacturers Association: *Confirmation of Approval for MIDI Standard: (CA-033) MIDI 1.0 Electrical Specification Update*. 2014, [Online; navštíveno 12.5.2019].
URL <https://www.midi.org/specifications>
- [10] MIDI Manufacturers Association: *The Complete MIDI 1.0 Detailed Specification*. 2014, [Online; navštíveno 14.1.2019].
URL <https://www.midi.org/specifications>

- [11] Oliver Hamann: *MACKIE HUI MIDI protocol*. 2010, [Online; navštíveno 10.3.2019].
URL <https://stash.reaper.fm/12332/HUI.pdf>
- [12] PreSonus Audio Electronics, Inc: *FaderPort*. 2019, [Online; navštíveno 5.1.2019].
URL <https://www.presonus.com/products/FaderPort>
- [13] Romkey, J.: *Nonstandard for transmission of IP datagrams over serial lines: SLIP*.
RFC 1055, RFC Editor, June 1988.
URL <http://www.rfc-editor.org/rfc/rfc1055.txt>
- [14] Savage, S.: *The Art of Digital Audio Recording: A Practical Guide for Home and Studio*. Oxford University Press, Inc., 2011, ISBN 978-0-19-539409-2.
- [15] Solid State Logic: *Nucleus²*. 2018, [Online; navštíveno 5.1.2019].
URL <https://www.solidstatelogic.com/studio/nucleus2>
- [16] The MIDI Association: *RTP-MIDI or MIDI over Networks*. Listopad 2015, [Online; navštíveno 10.1.2019].
URL <https://www.midi.org/articles/rtp-midi-or-midi-over-networks>
- [17] Wikipedia contributors: Digital audio workstation — Wikipedia, The Free Encyclopedia. 2019, [Online; navštíveno 14.1.2019].
URL https://en.wikipedia.org/w/index.php?title=Digital_audio_workstation&oldid=878380495

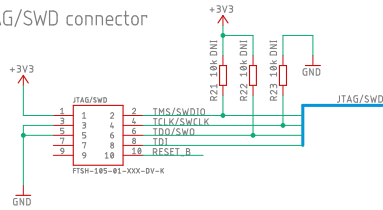
Příloha A

Schéma Fader modulu

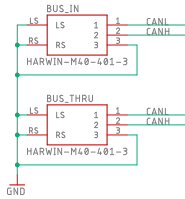
Schéma Fader modulu je rozděleno do několika stran:

- **strana 56:** rozhraní modulu - konektory
- **strana 57:** napájení – spínaný zdroj, LDO regulátor pro ADC, napájení MCU
- **strana 58:** zapojení mikrokontroléru a CAN transceiveru
- **strana 59:** senzory dotyku pro jezdcy potenciometrů
- **strana 60:** AD převodníky s napěťovými referencemi
- **strana 61:** H-můstky
- **strana 62:** zapojení ledbarů

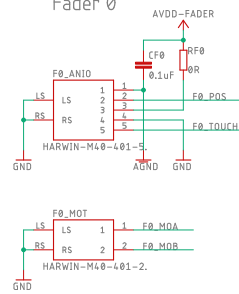
JTAG/SWD connector



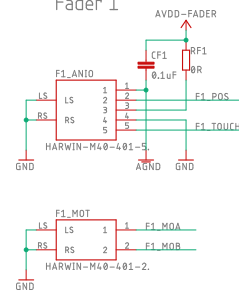
Main bus connectors



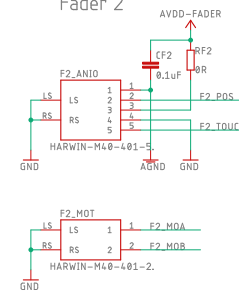
Fader 0



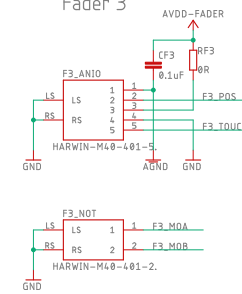
Fader 1



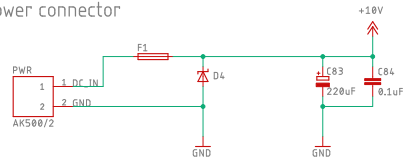
Fader 2



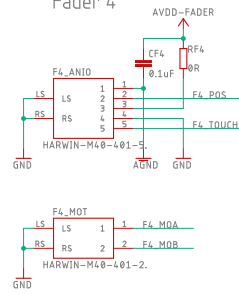
Fader 3



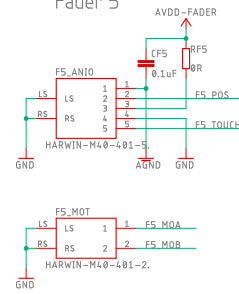
Power connector



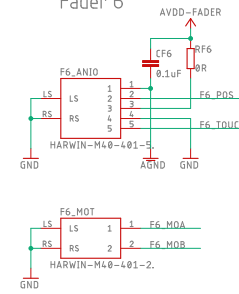
Fader 4



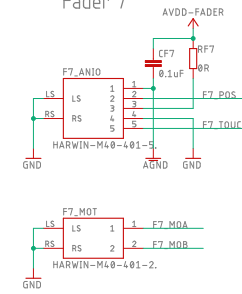
Fader 5



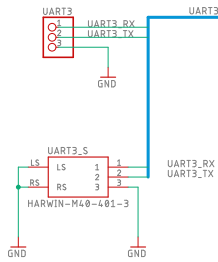
Fader 6



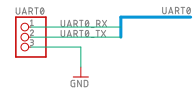
Fader 7



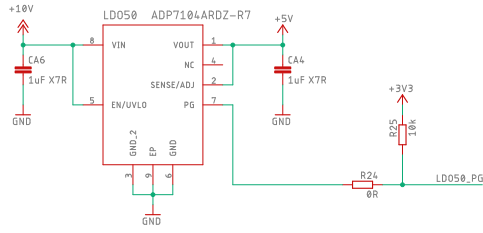
UART3



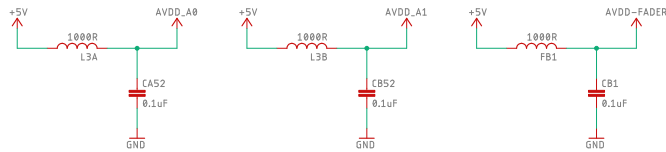
UART0



+5V LDO regulator for ADCs

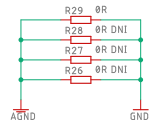


Filtering +5V for each ADC

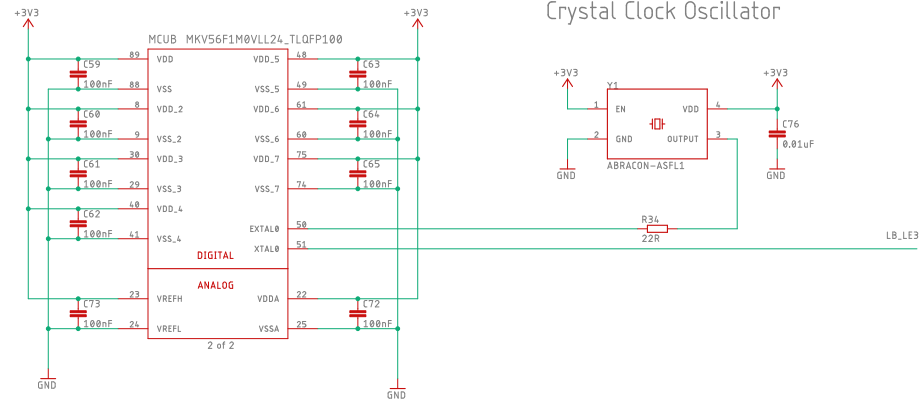


Connect AGND with GND

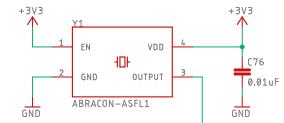
Possible ferrite bead



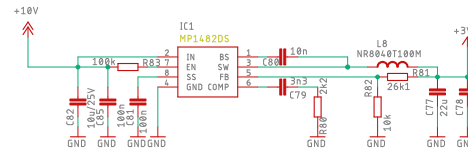
MCU power supply

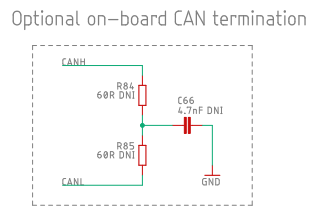
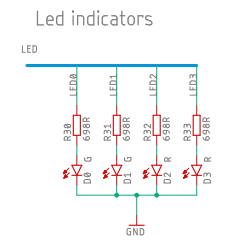
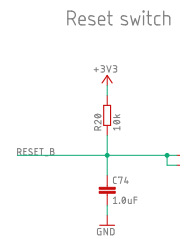
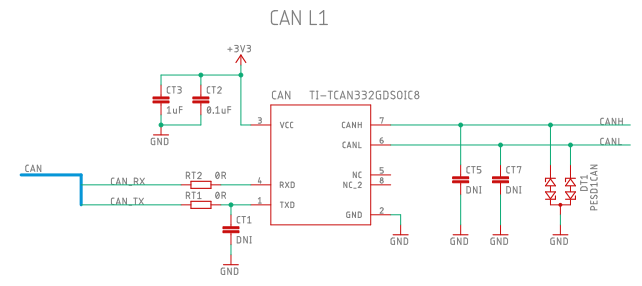
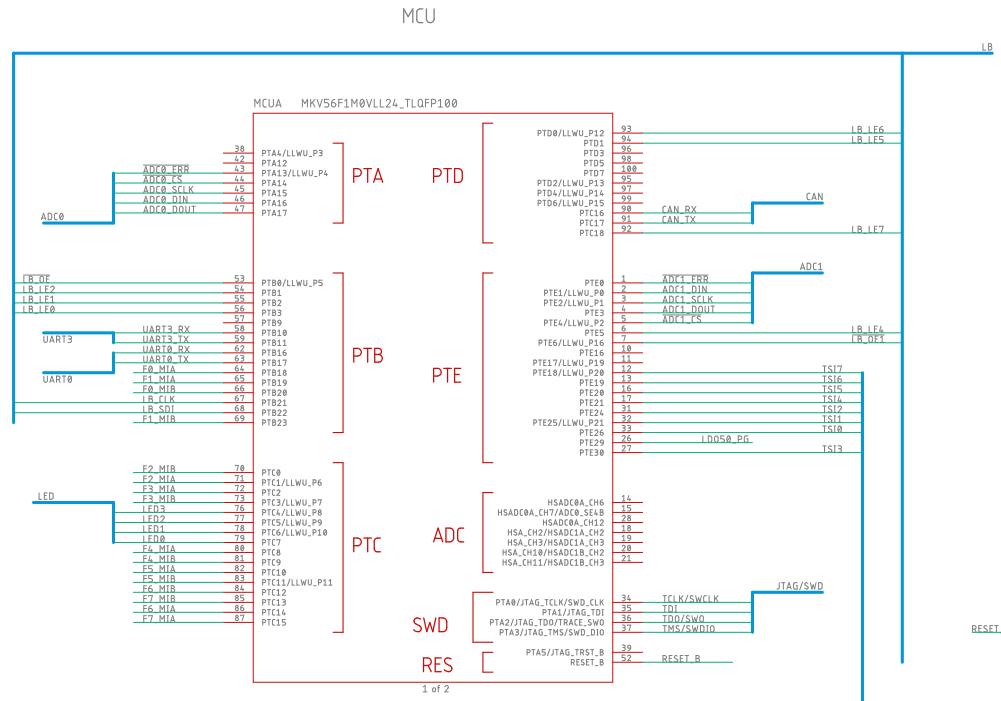


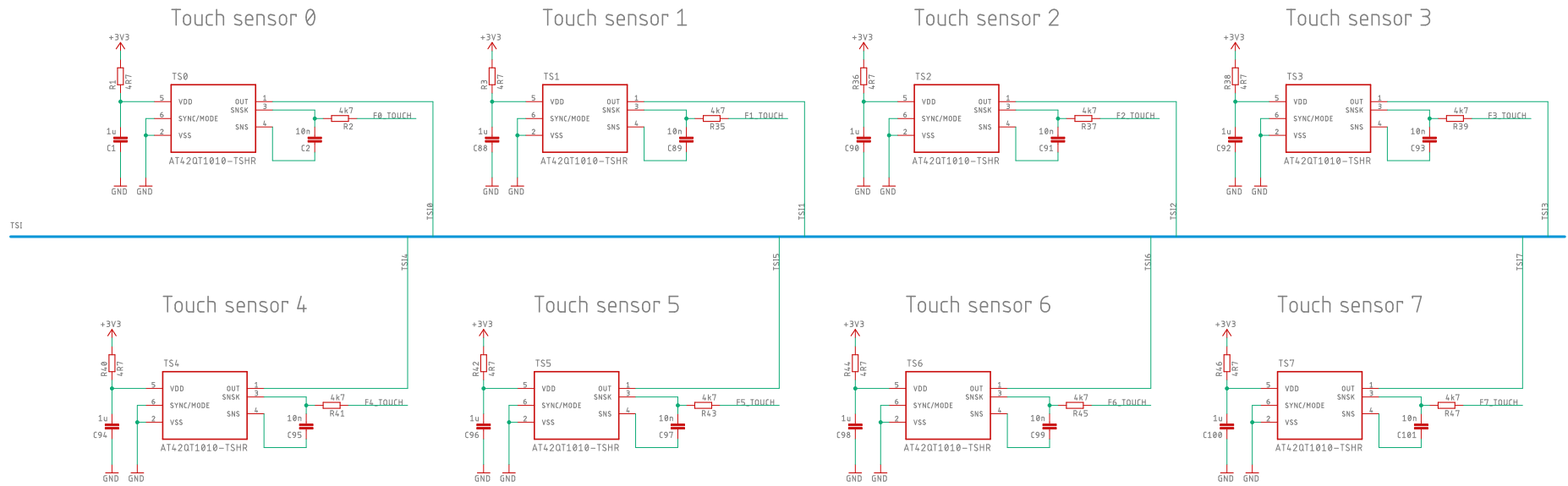
Crystal Clock Oscillator



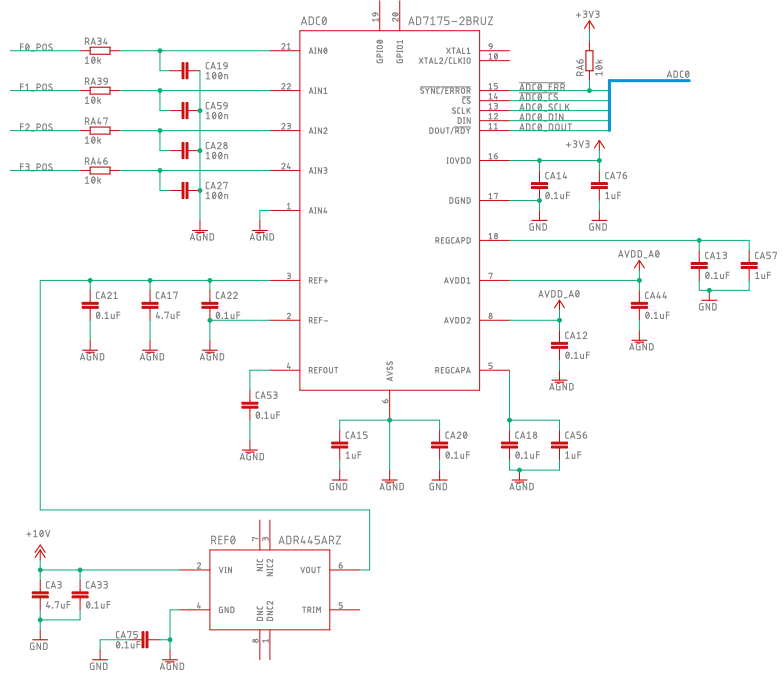
Switched main power supply



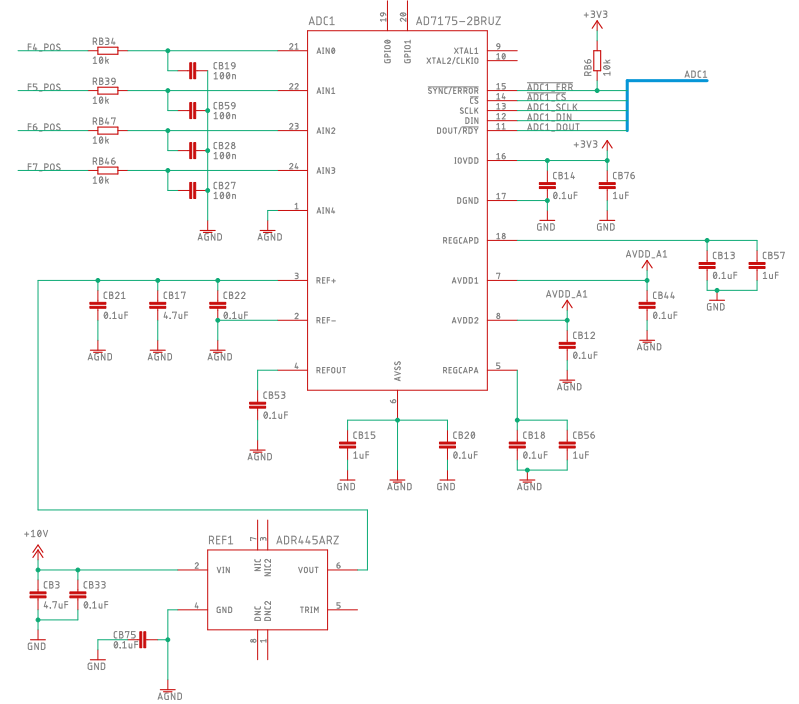




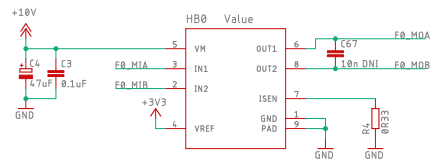
ADC0



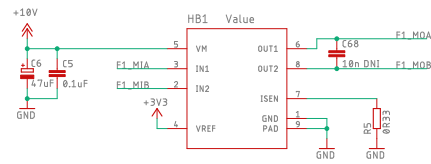
ADC1



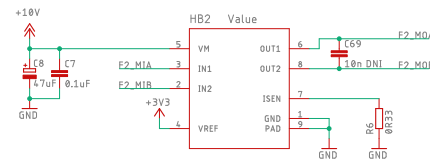
H-bridge 0



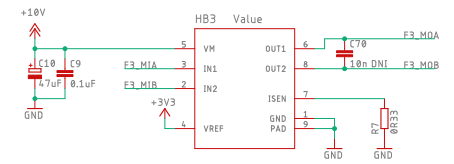
H-bridge 1



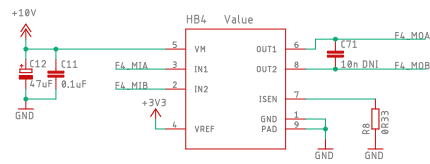
H-bridge 2



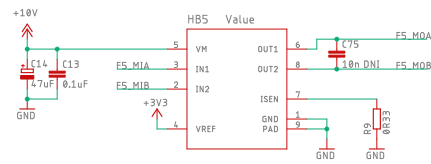
H-bridge 3



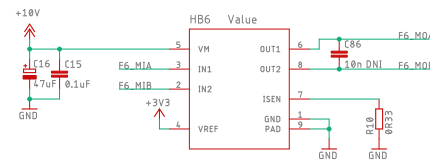
H-bridge 4



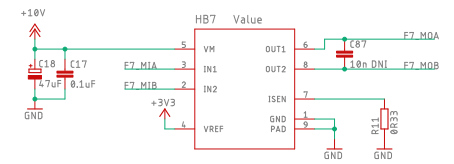
H-bridge 5

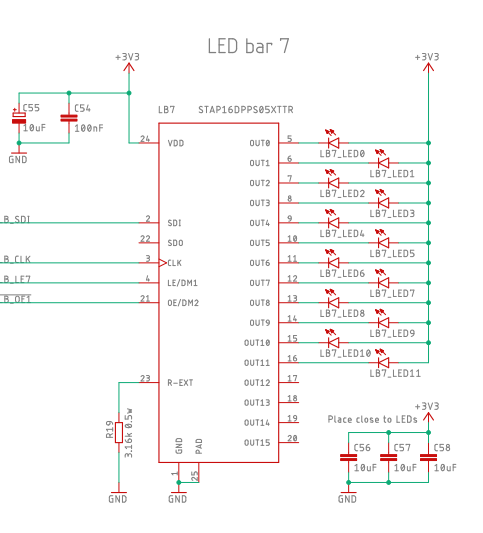
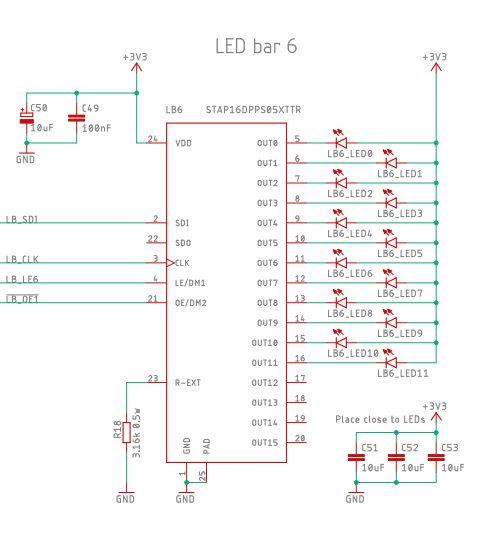
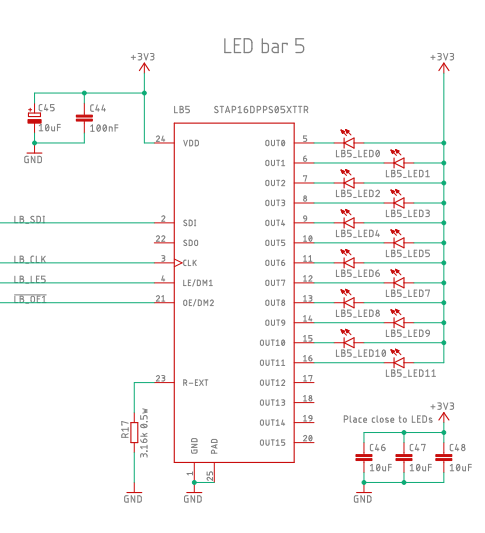
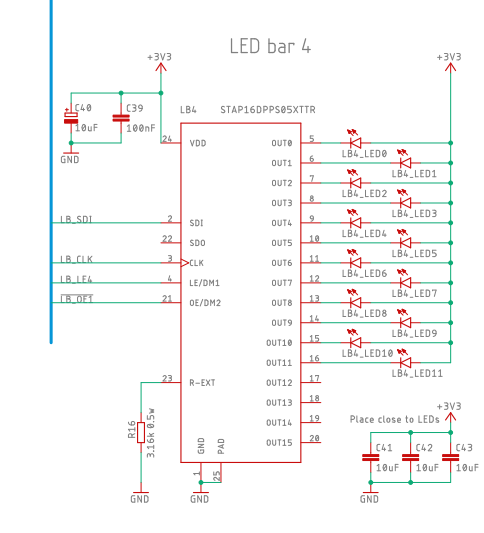
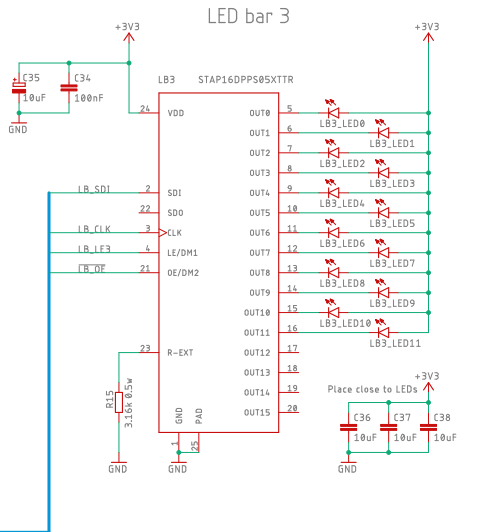
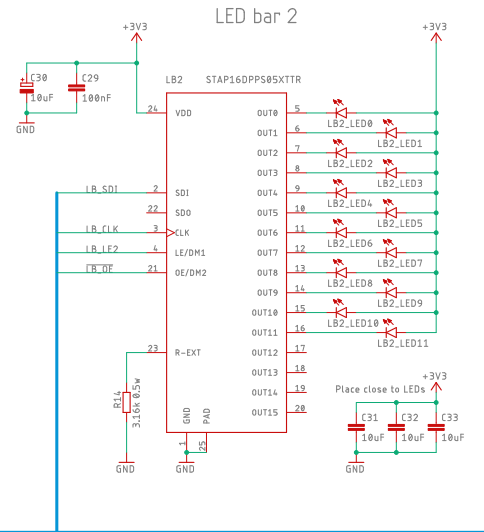
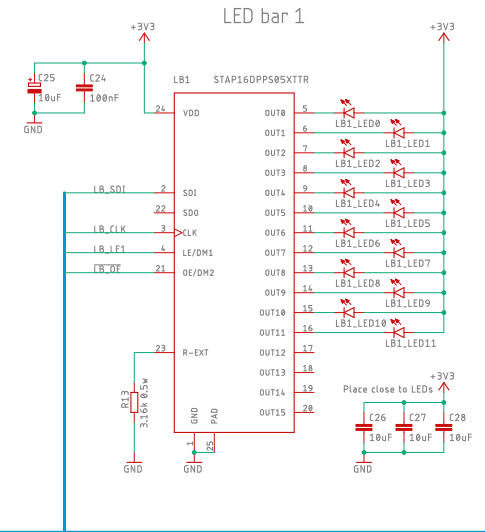
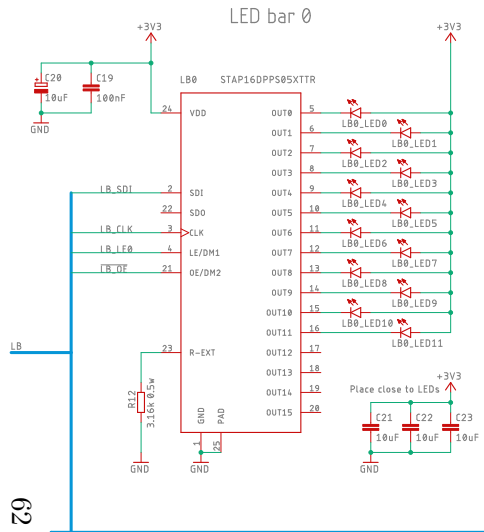


H-bridge 6



H-bridge 7



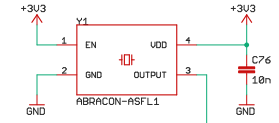
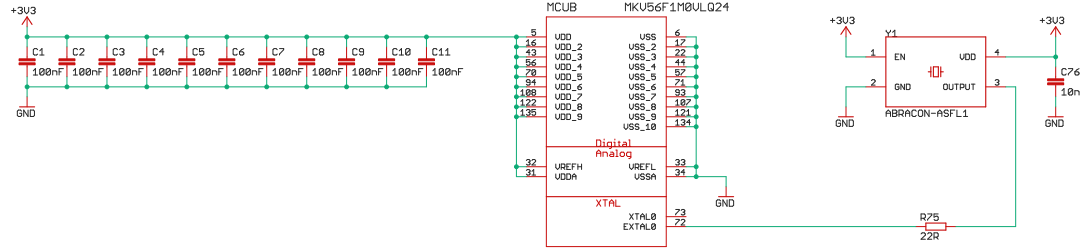
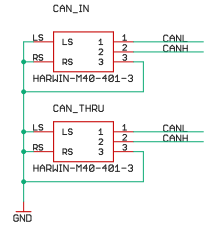
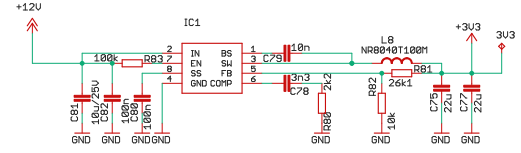
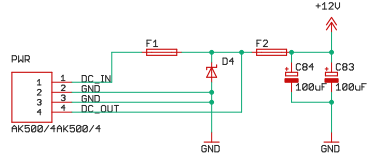
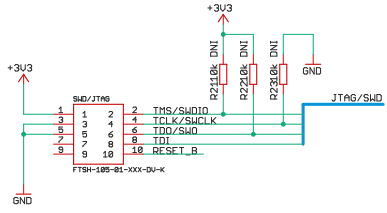


Příloha B

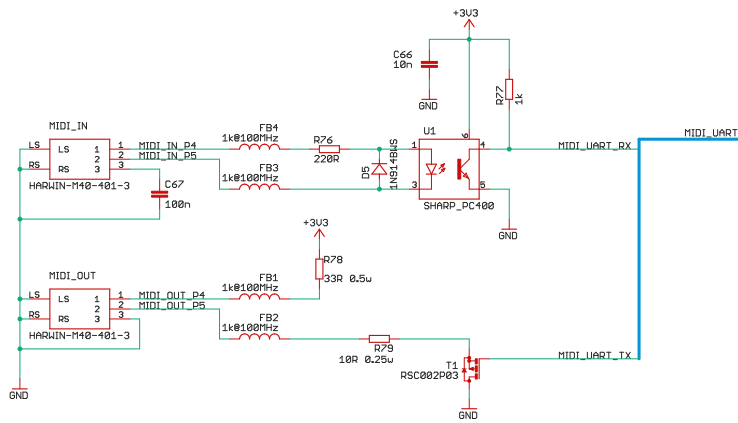
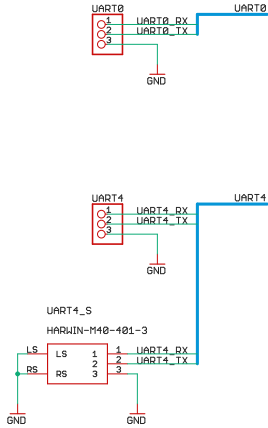
Schéma Line modulu

Schéma Line modulu je rozděleno do několika stran:

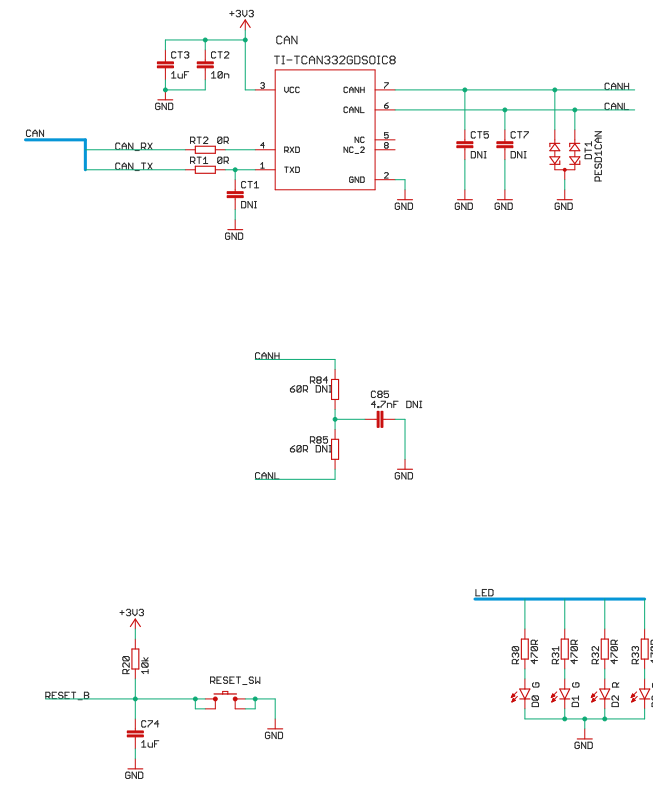
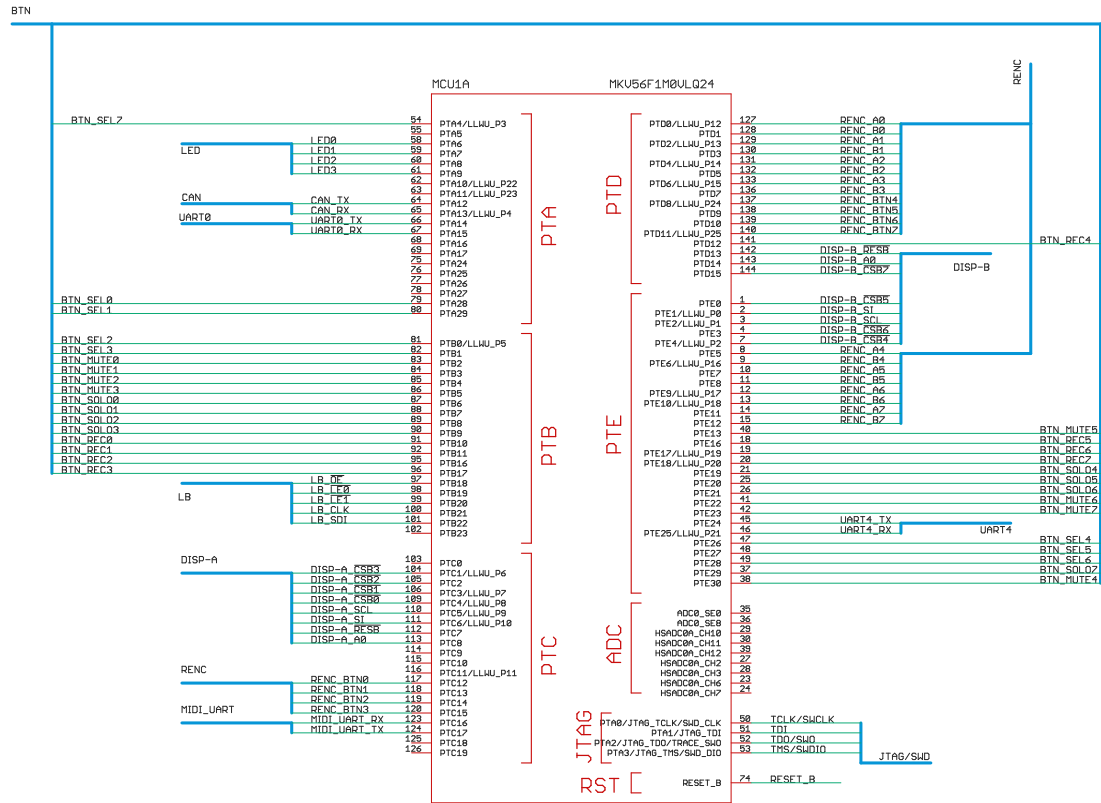
- **strana 64**: konektory modulu, spínaný zdroj, napájení MCU, oddělovací MIDI obvod
- **strana 65**: zapojení mikrokontroléru a CAN transceiveru
- **strana 66**: zapojení podsvícených tlačítek
- **strana 67**: LED drivery
- **strana 68**: zapojení konektorů pro displeje
- **strana 69**: rotační enkodéry



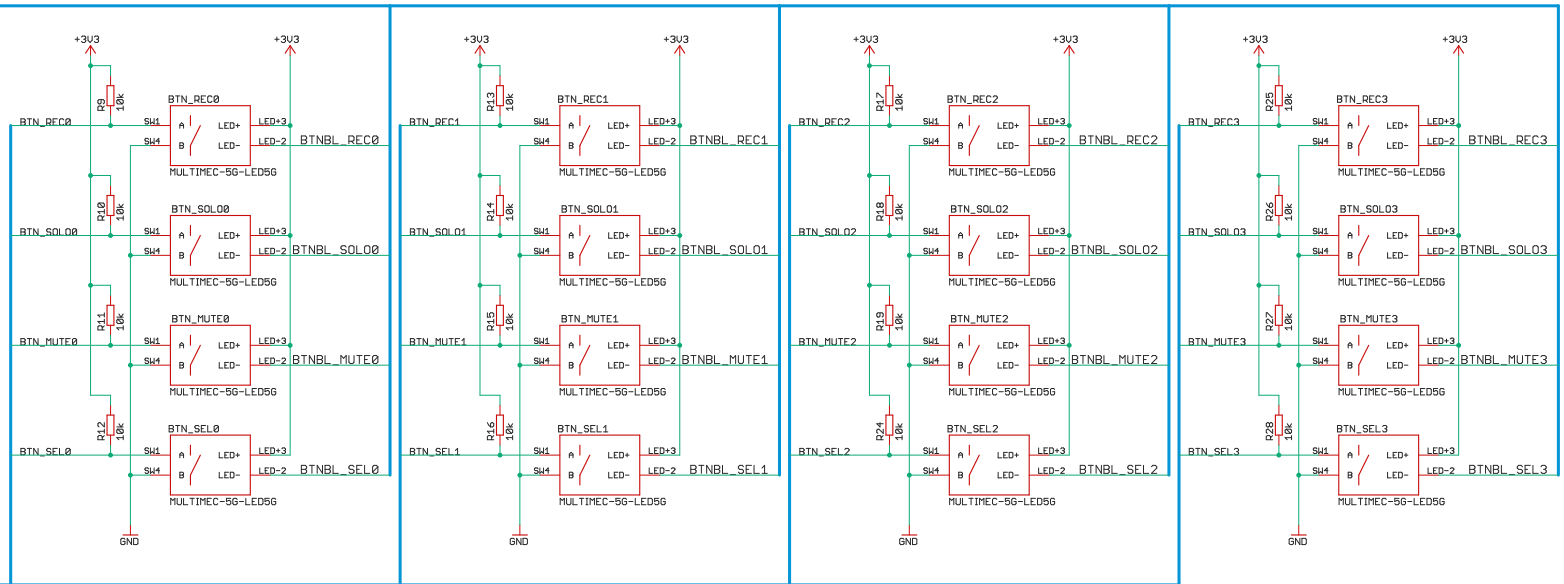
64



65

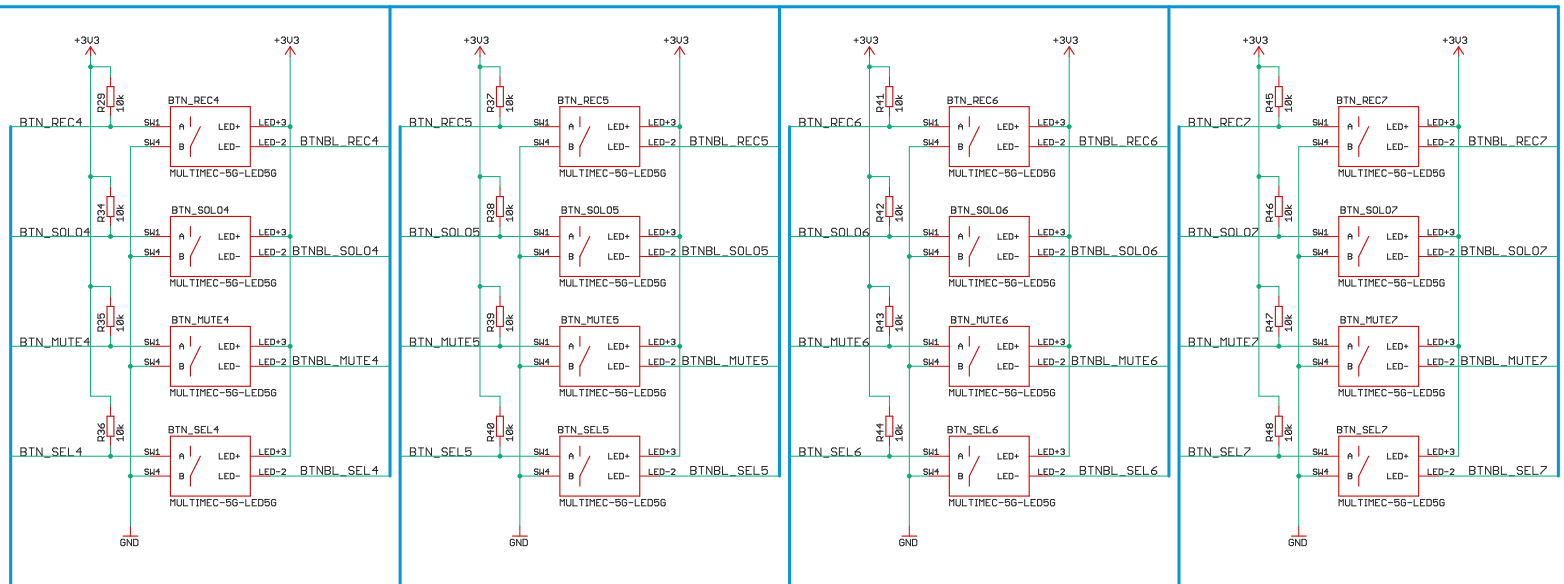


BTNBL

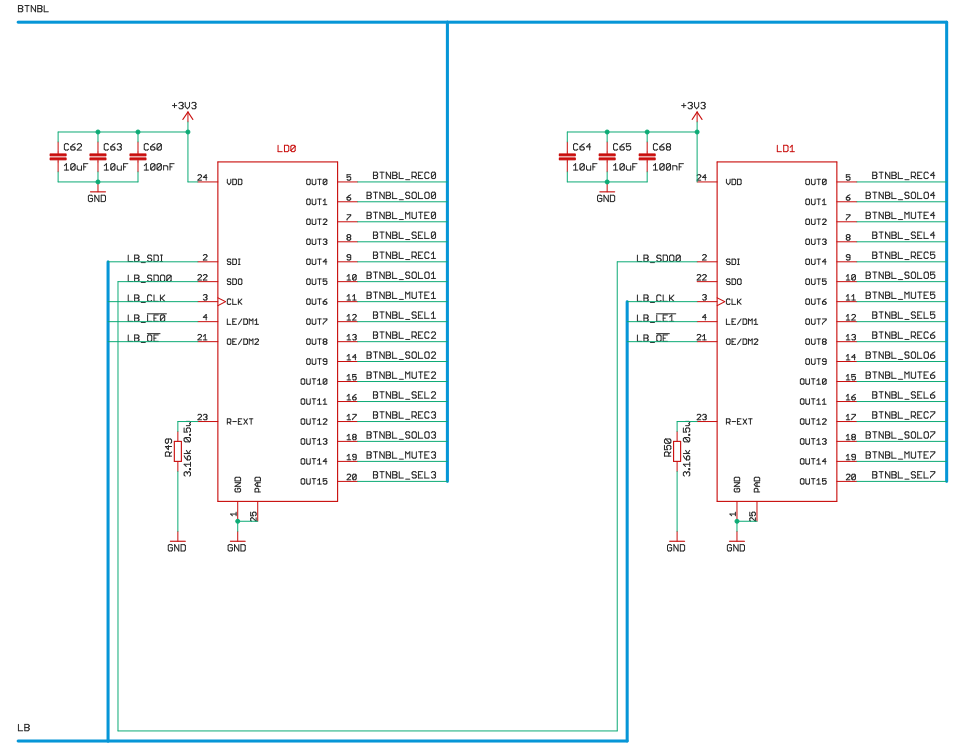


BTN

BTNBL



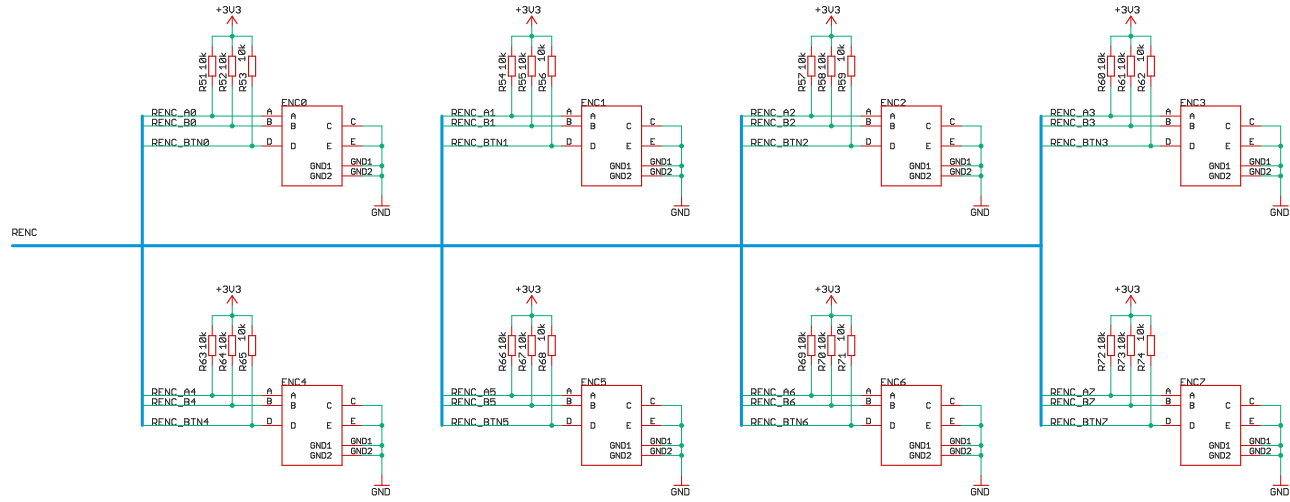
BTN



BTNBL

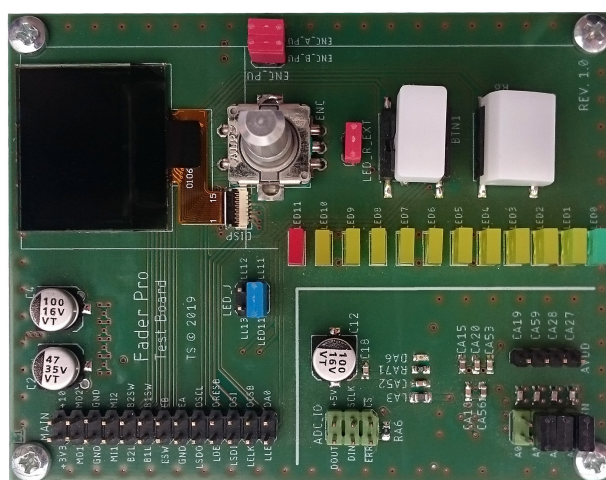
LB



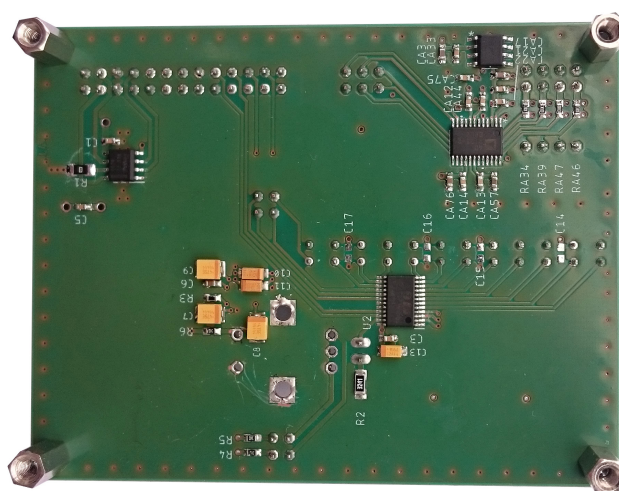


Příloha C

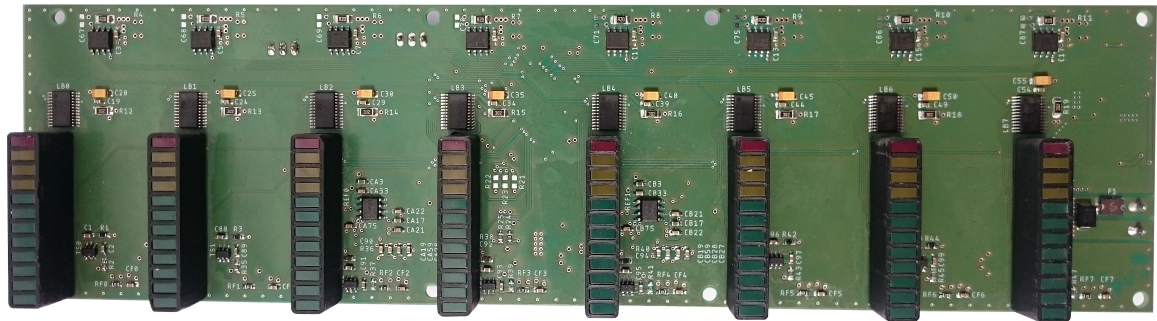
DPS vytvořených modulů



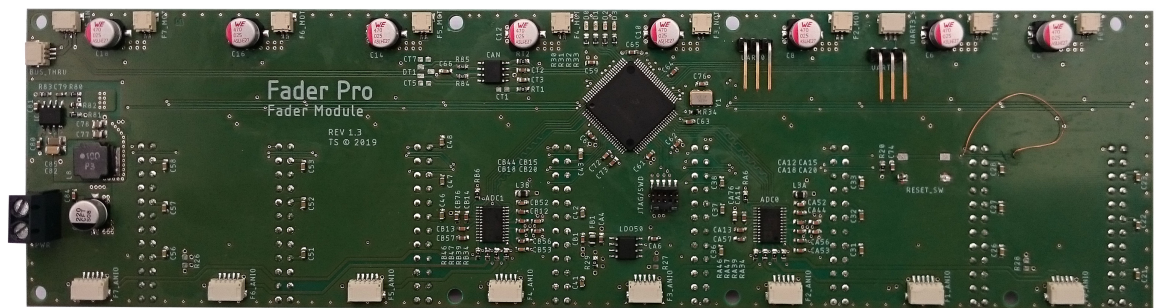
Obrázek C.1: Vrchní strana osazené testovací desky



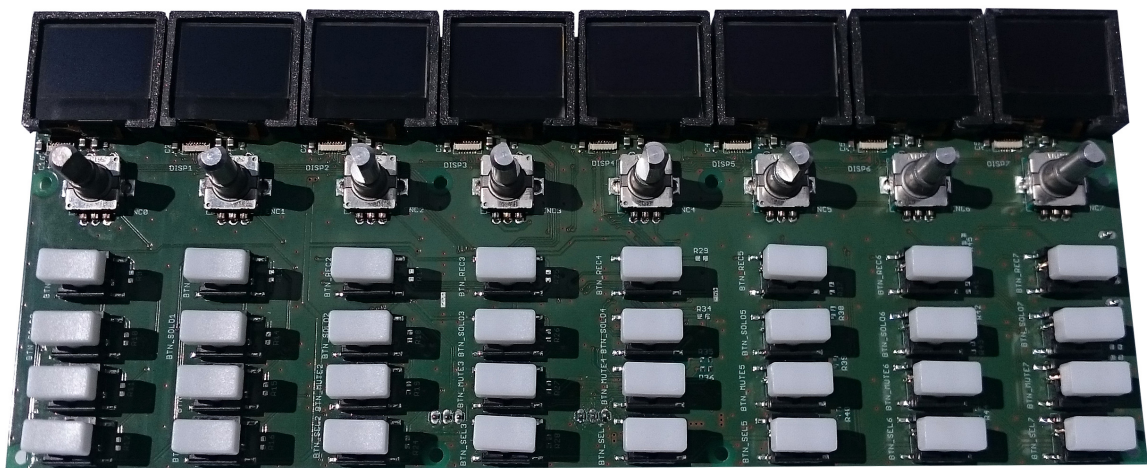
Obrázek C.2: Spodní strana osazené testovací desky



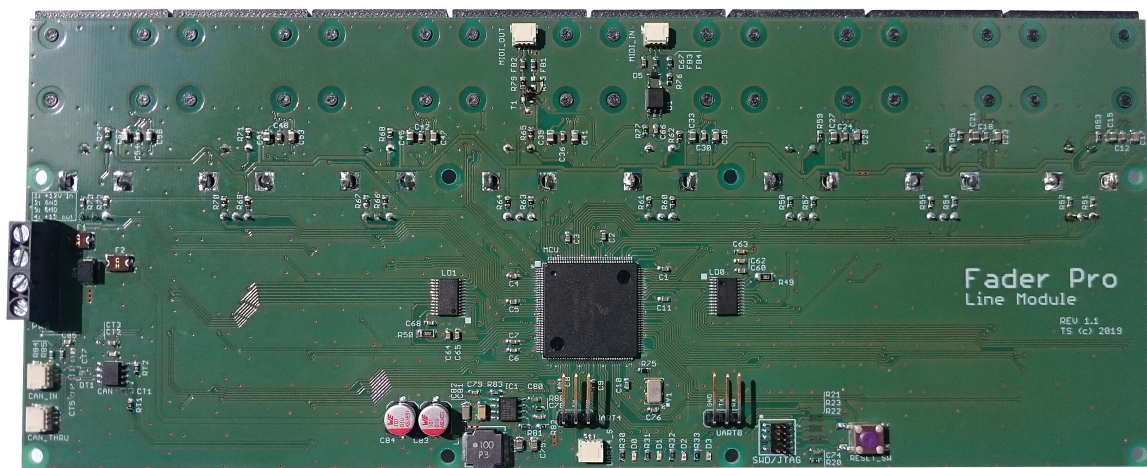
Obrázek C.3: Vrchní strana osazené DPS Fader modulu



Obrázek C.4: Spodní strana osazené DPS Fader modulu



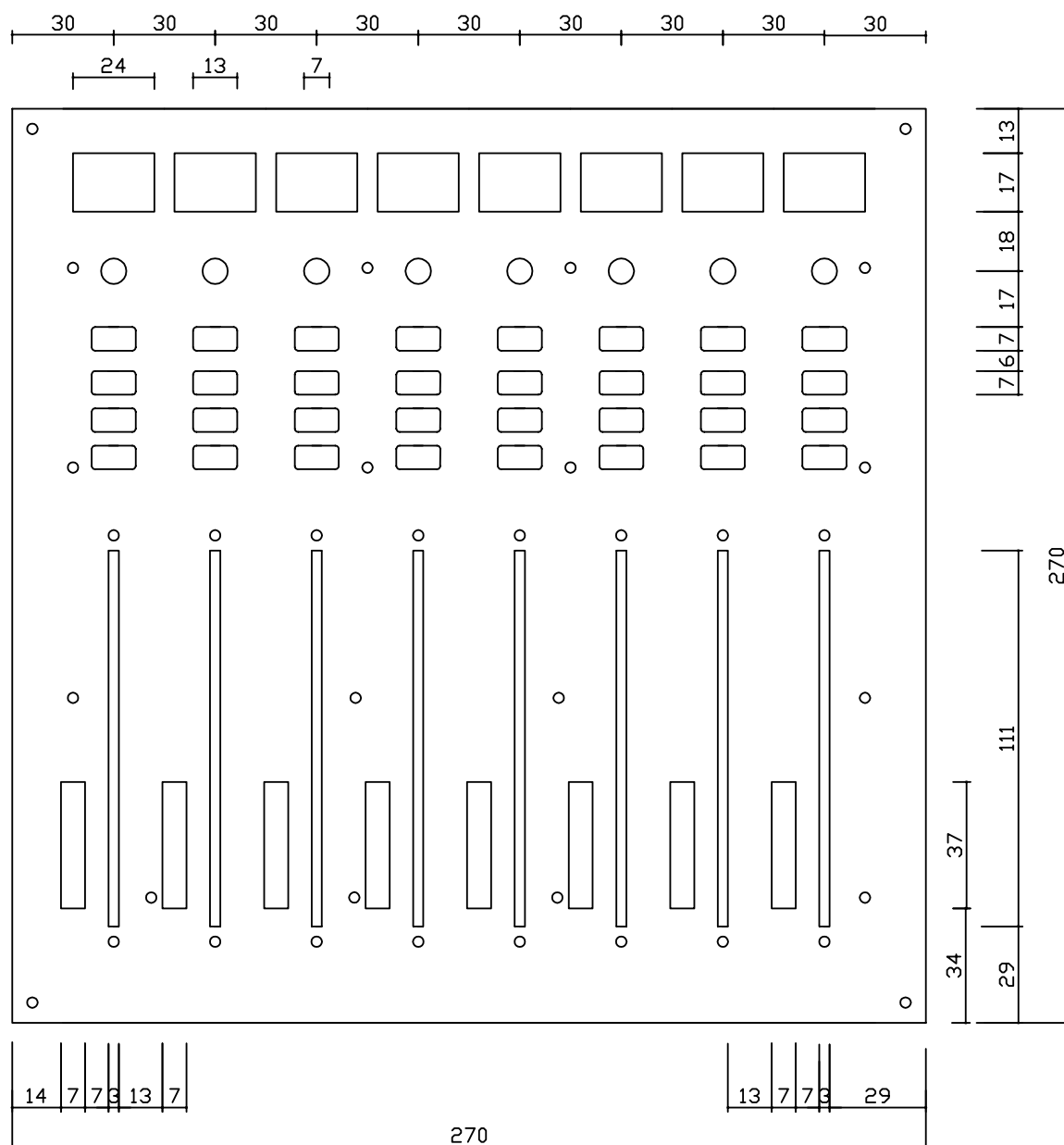
Obrázek C.5: Vrchní strana osazené DPS Line modulu



Obrázek C.6: Spodní strana osazené DPS Line modulu

Příloha D

Nákres panelu zařízení



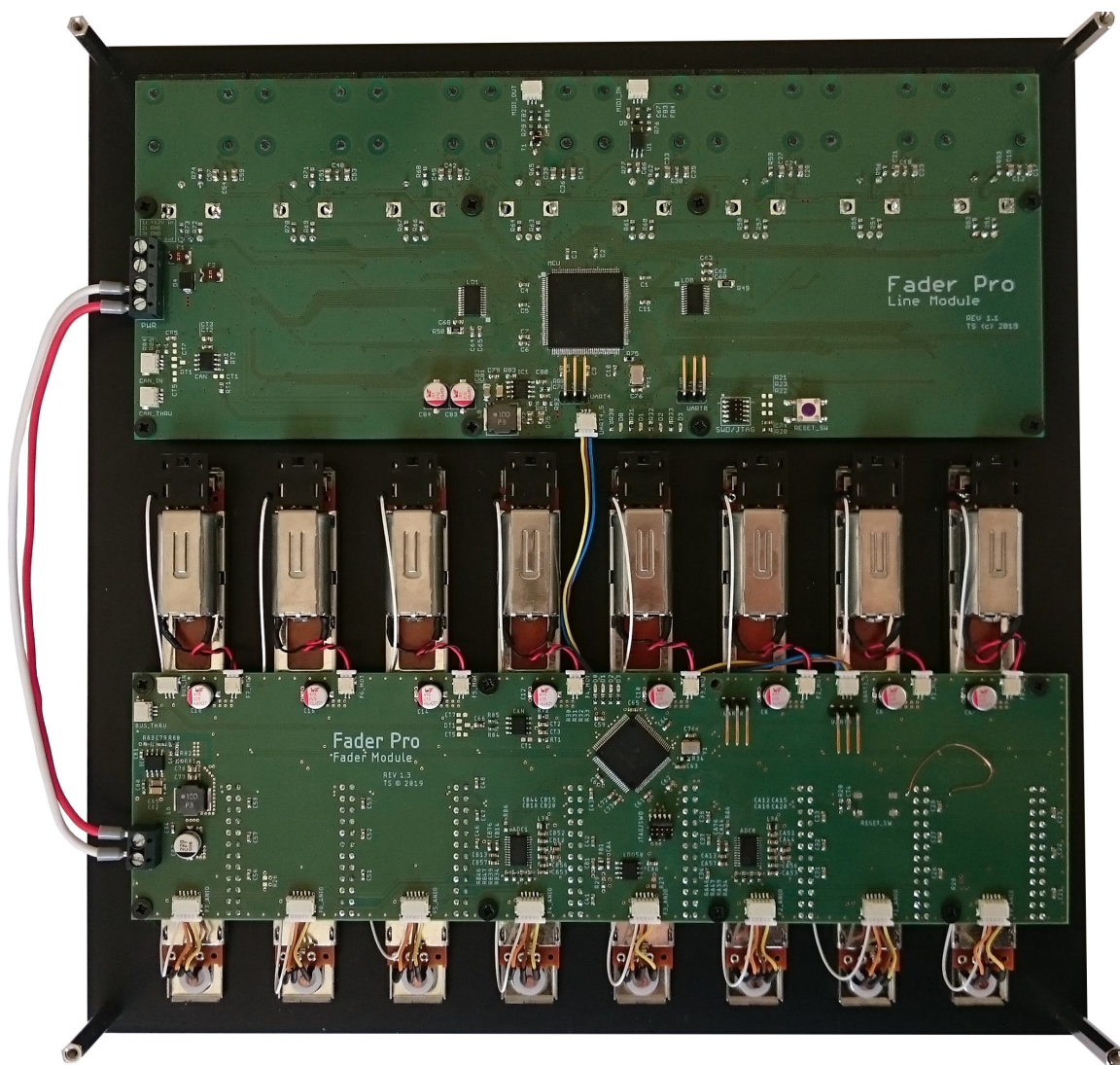
Obrázek D.1: Nákres hliníkového panelu vytvořeného zařízení

Příloha E

Vyrobené zařízení



Obrázek E.1: Fotografie prototypu DAW kontroléru, který je výsledkem této práce (z pohledu ovládacích prvků)



Obrázek E.2: Fotografie prototypu DAW kontroléru, který je výsledkem této práce (z pohledu DPS)