

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra kybernetiky

Využití robota LEGO Mindstorms EV3 - návrh robota hrajícího na piano pro propagaci FEL

Martin Němec

Vedoucí práce: Ing. Martin Hlinovský, Ph.D.
Studijní program: Kybernetika a robotika
Obor: Robotika
Květen 2018

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Němec** Jméno: **Martin** Osobní číslo: **457200**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra kybernetiky**
Studijní program: **Kybernetika a robotika**
Studijní obor: **Robotika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Využití robota LEGO Mindstorms EV3 - návrh robota hrajícího na piano pro propagaci FEL

Název bakalářské práce anglicky:

Usage of the LEGO Mindstorms EV3 - Design of the Piano Playing Robot for Promotion of the Faculty

Pokyny pro vypracování:

1. Seznamte se s možnostmi robota Lego Mindstorms EV3 (současný stav, HW a SW vybavení).
2. Proveďte a realizujte návrh robota hrajícího na piano pro propagační účely FEL (např. <https://www.youtube.com/watch?v=TDvcTa3abRM>)
3. Vytvořte webové stránky k realizovanému projektu (popis, návrh regulátoru nebo principu činnosti, vysvětlení navrženého softwaru, fotogalerii a popřípadě návod na stavbu robota).
4. Popřípadě navrhnete software pro převod písniček z formátu MIDI do souboru nahratelného přímo do kostky EV3.

Seznam doporučené literatury:

- [1] James Floyd Kelly - LEGO MINDSTORMS NXT-G programming Guide, Second Edition
- [2] Daniele Benedettelli - Programming LEGO NXT Robots using NXC
- [3] <https://www.youtube.com/watch?v=TDvcTa3abRM>
- [4] <https://www.youtube.com/watch?v=TImEpyKwpl>

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Martin Hlinovský, Ph.D., katedra řídicí techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **12.01.2018**

Termín odevzdání bakalářské práce: **25.05.2018**

Platnost zadání bakalářské práce: **30.09.2019**

Ing. Martin Hlinovský, Ph.D.
podpis vedoucí(ho) práce

doc. Ing. Tomáš Svoboda, Ph.D.
podpis vedoucí(ho) ústavu/katedry

prof. Ing. Pavel Ripka, CSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval vedoucímu mé bakalářské práce Ing. Martinovi Hlinovskému, Ph.D. za konzultace a vstřícný přístup.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 25. května 2018

.....

Abstrakt

Tato práce je zaměřena na konstrukci a programování robota hrajícího na piano za použití stavebnice Lego Mindstorms EV3. Program robota s názvem PianoPlay je psán v jazyce Java a je spuštěn v prostředí LeJOS EV3. Dále je vytvořen program pro osobní počítač, též v jazyce Java, určený ke konverzi hudebních souborů formátu MIDI na formát podporovaný programem PianoPlay.

Součástí práce je i tvorba webové stránky s návody pro stavbu a obsluhu robota určená veřejnosti.

Klíčová slova: Lego Mindstorms EV3, LeJOS, Java, MIDI, PianoPlay

Vedoucí práce: Ing. Martin Hlinovský, Ph.D.

Abstract

This thesis focuses on the construction and programming of a piano playing robot, using the Lego Mindstorms EV3 kit. The program of the robot called PianoPlay is written in Java programming language and is executed under LeJOS EV3 environment. Additionally, a program for a PC, also written in Java, is created, which converts music files in MIDI format into another format supported by PianoPlay.

Part of the thesis is also the creation of a website containing instructions for construction and operation of the robot, intended for the public.

Keywords: Lego Mindstorms EV3, LeJOS, Java, MIDI, PianoPlay

Title translation: Usage of the LEGO Mindstorms EV3 - Design of the Piano Playing Robot for Promotion of the Faculty

Obsah

1 Úvod	1	3.2.1 Řídící program robota - PianoPlay	14
2 Mechanická konstrukce	3	3.2.2 Konverzní program pro PC - PianoPlay converter	18
2.1 Výběr komponent	3	4 Webová stránka	21
2.1.1 Klávesy	3	4.1 Lego Digital Designer	21
2.1.2 Lego Mindstorms EV3	4	5 Závěr	23
2.2 Stavba	6	A Literatura	25
2.2.1 PianoPlayer	6	B Obsah příloženého CD	29
2.2.2 Sustain controller	9		
2.2.3 Stolek	10		
3 Programy	11		
3.1 Použitý software	11		
3.1.1 Java	11		
3.1.2 LeJOS EV3	12		
3.1.3 MIDI	12		
3.1.4 MidiEditor	13		
3.2 Vlastní software	13		

Obrázky

2.1 Klávesy CASIO CTK-2400 [1] . . .	4	4.1 Grafické uživatelské rozhraní programu Lego Digital Designer [4]	22
2.2 Inteligentní kostky Mindstorms: RCX (vlevo), NXT (uprostřed), EV3 (vpravo) [2]	6	5.1 Finální verze robota se stolkem a elektrickými klávesami	24
2.3 Modul robota, pohled shora	7		
2.4 První typ aktuátoru	7		
2.5 Druhý typ aktuátoru	8		
2.6 Třetí typ aktuátoru	8		
2.7 Otočený modul robota, pohled na aktuátory	9		
2.8 Sustain controller přichycený ke stolku	10		
2.9 Stolek pro robota; pohled shora (nahore), pohled zepředu (dole) . . .	10		
3.1 Grafické uživatelské rozhraní programu MidiEditor [3]	14		
3.2 Uživatelské rozhraní robota - Výběr postavení třídy <i>MainMenu</i> (vlevo), výběr skladby třídy <i>Master</i> (uprostřed), schéma ovládaných tónů třídy <i>Player</i> (vpravo)	18		
3.3 Grafické uživatelské rozhraní programu PianoPlay converter	19		



Kapitola 1

Úvod

Na přelomu devatenáctého a dvacátého století se začala rozšiřovat samohrající piana (tzv. pianoly) na veřejnosti i v domácnostech [5]. Pianola vznikla přidáním pneumatického ovládacího mechanismu do klasického piana. Hudba pro pianolu je naprogramována v podobě děrovaného papíru (obdoba děrných pásek). Každému tónu odpovídá jeden řádek na programovém papíru. Hra na pianolu spočívá pouze v pumpování vzduchu pomocí dvou pedálů. Díky tomu je vytvářen podtlak v zásobníku. Dírou v programovém papíru je nasáván vzduch, který následně zapříčiní stisk klávesy piana. Koncem dvacátých let dvacátého století byla výroba pianol kvůli nízkému zájmu veřejnosti z důvodu elektrického záznamu a reprodukce hudby ukončena. V současnosti se vyrábí elektromechanické pianoly ovládané digitálně, většinou s využitím hudebního standardu MIDI.

Fakulta elektrotechnická využívá ke své prezentaci na různých výstavách mimo jiné i roboty sestavené ze stavebnice Lego Mindstorms, například sledovač černé čáry nebo skladač Rubikovy kostky. K upoutání pozornosti návštěvníků by mohla sloužit hudba. Proto vznikla myšlenka vytvořit robota hrajícího na piano. Existuje mnoho pokusů o stavbu takového robota ze stavebnice Lego. Většina z nich je ale buď specificky stavěná pro určitou skladbu [6], nebo v případě univerzálního robota sice zvládne zahrát kteroukoli skladbu, ale nastávají ve skladbách prodlevy při posunu aktuátorů po klávesnici [7]. Cílem práce tedy bylo vytvořit ze stejné stavebnice robota, který zvládne plynule zahrát libovolnou skladbu načtenou ze souboru.

Kapitola 2

Mechanická konstrukce

V první části této kapitoly je podrobně popsán použitý hardware. Druhá část je pak zaměřena na konstrukci robota a jeho příslušenství.

2.1 Výběr komponent

Dle zadání byla ke konstrukci využita robotická stavebnice Lego Mindstorms EV3 obsahující všechny potřebné komponenty ke stavbě robota. Stavebnice byla zapůjčena školou, která ji využívá v motivačním předmětu Roboti, kde studenti staví robota určeného pro specifickou úlohu a následně spolu soupeří. Tato úloha je zároveň zadáním Robosoutěže [8], což je akce určená pro studenty základních a středních škol organizovaná Katedrou řídicí techniky.

2.1.1 Klávesy

Aby mohl robot hrát, potřebuje hudební nástroj. V zadání je uvedeno piano, nicméně klasické piano forte, nebo pianino by bylo pro transport velmi nevhodné. Z toho důvodu byly vybrány elektronické klávesy CASIO CTK-2400 [1] vyhovující jak rozměry, hmotností, tak i cenou. Klávesy mají rozsah pět oktáv – od tónu C2 po C7 s 48-hlasou polyfonií, tedy dokáží hrát najednou maximálně 48 tónů. Pomocí konektoru typu Jack 6.3 mm lze připojit externí

reproduktory a zvýšit tak hlasitost přehrávání. Dalším konektorem Jack 6.3 mm je možné připojit tzv. sustain (nebo též damper) pedál, který umožňuje prodloužit zahrané zvuky tónů, aniž by musely být nadále stisknuty odpovídající klávesy.

Tyto elektronické klávesy nemají dynamiku stisku, což je další výhoda oproti klasickému pianu, neboť dynamika stisku by byla velmi obtížně simulována aktuátory z použité stavebnice, a tóny by tak zněly příliš hlasitě a ostře.



Obrázek 2.1: Klávesy CASIO CTK-2400 [1]

Klávesy piana se seskupují do tzv. oktáv, stejně jako samotné noty. Oktáva, jak název napovídá, je skupina osmi not, kde následující nota je vždy o tón, nebo půltón vyšší než předcházející [9]. Zpravidla oktáva začíná tónem C, následují tóny D, E, F, G, A, H a poslední tón je C' (s dvojnásobnou frekvencí než C). Tyto tóny jsou na klaviatuře reprezentovány bílými klávesami. Některé tóny lze rozdělit i na půltóny značené Cis, Dis, Fis, Gis, Ais (nebo též C \sharp , D \sharp , F \sharp , G \sharp , A \sharp), reprezentované černými klávesami klaviatury. Každá oktáva je tedy tvořena dvanácti klávesami, třináctá klávesa C' se počítá do následující oktávy.

■ 2.1.2 Lego Mindstorms EV3

Pro konstrukci robota byla vybrána robotická stavebnice Lego Mindstorms EV3. Je to v současné době nejnovější verze stavebnice Mindstorms, což je stavebnice typu Lego Technic, která kromě klasických plastových stavebních dílů Lego obsahuje také elektronické prvky – akční členy, senzory a řídicí jednotku (někdy také označovanou jako inteligentní kostka).

První verzí Mindstorms byla stavebnice RIS (Robotics Invention System) vydaná roku 1998 [10, 11]. Tato stavebnice obsahovala inteligentní kostku zvanou RCX (Robotic Command Explorer) jejímž základem byl osmibitový procesor H8/300 firmy Hitachi při taktu 16 MHz s 16kB programové paměti a 32 kB operační paměti. Kostka byla dále vybavena infračerveným portem pro komunikaci, malým segmentovým LCD displejem, třemi porty pro ovládání motorů a třemi porty pro senzory, jimiž byla tlačítka a světelný senzor. Díky rychlému pokroku v oblasti elektrotechniky byla však tato revoluční stavebnice překonána svými nástupci.

Po osmi letech od vydání první inteligentní kostky byla v roce 2006 zařazena do prodeje následující generace s názvem Lego Mindstorms NXT (NeXT – další generace) [12, 13]. Inteligentní kostka je plně přepracována. Jádrem systému je dvaatřicetibitový procesor rodiny ARM7 od firmy Atmel při taktu 48 MHz s 256 kB programové Flash paměti a 64 kB operační paměti. Pro komunikaci s počítačem využívá kostka USB port, nebo spojení Bluetooth. Uživatelské rozhraní tvoří LCD displej o velikosti 100 x 64 pixelů, 4 tlačítka sloužící k pohybu v menu, či ovládání programu a malý reproduktor, který je velmi užitečný při ladění programu. Kostka je dále vybavena třemi porty umožňujícími ovládání servomotorů a čtyřmi porty určenými pro připojení senzorů pomocí sběrnice I2C. Mezi senzory náležící do základního setu patří tlačítka, senzor světla, ultrazvukový dálkoměr a mikrofon, nicméně díky zveřejnění použitého hardwaru (Hardware Developer Kit) jsou vyráběny další různé senzory jinými firmami (např. firma HiTechnic [14] vyrábí akcelerometr, gyroskop, barometrický senzor, senzor mechanického otočení, kompas . . .).

Při patnáctém výročí stavebnice Mindstorms byla v roce 2013 představena nejnovější verze EV3 (Evolution 3) [13, 15]. Stavebnice EV3 a NXT jsou velmi podobné, ačkoli mají jiný design plastových krytů. V nové verzi navíc přibyl servomotor střední velikosti, gyroskopický senzor a senzor světla byl nahrazen senzorem barev. Nejrozsáhlejší změny byly provedeny u řídicí jednotky, tedy inteligentní kostky, které jdou ruku v ruce s neustálým vývojem elektroniky. Kostka využívá procesor rodiny ARM9 firmy Texas Instruments při taktu 300MHz, který spolu s čtečkou paměťových karet umožňuje běh upravené verze operačního systému Linux. Paměť kostky byla změněna následovně: operační paměť 64 MB, interní programová Flash paměť 16 MB plus čtečka paměťových karet microSDHC až do 32 GB. Do uživatelského rozhraní přibyla dvě tlačítka (celkem má kostka tedy šest tlačítek), rozlišení LCD displeje bylo zvětšeno na 178 x 128 pixelů a reproduktor byl vyměněn za kvalitnější. Kostka EV3 také obsahuje USB Host port, který otevírá nové možnosti, ať už připojení kamery, nebo třeba Wi-Fi adaptéru. Rovněž byl přidán čtvrtý port pro ovládání servomotorů.



Obrázek 2.2: Inteligentní kostky Mindstorms: RCX (vlevo), NXT (uprostřed), EV3 (vpravo) [2]

2.2 Stavba

Nejčastější styl hry na piano je tzv. obouruční hra, tedy využití dvou rukou k obsluze klaviatury. To znamená dvě pětky aktuátorů, každá dokáže ovládat jisté okolí své pozice a každá může měnit svou pozici nezávisle na druhé. Tato technika však není pro konstrukci ze stavebnice Lego vhodná, protože Lego motory by nestíhaly měnit pozici aktuátorů dostatečně rychle a melodie by tak musela být rozkouskovaná a ztratila by rytmus [7, 16]. Z toho důvodu bylo rozhodnuto, že jednotlivé aktuátory budou staticky umístěny pro všechny tóny klaviatury.

2.2.1 PianoPlayer

Robot byl pojmenován PianoPlayer přesně podle své funkce. První otázka, která nastala, se týkala rozdílu rozteče stavebnicových dílů a pianových kláves. Naštěstí jedna pianová klávesa přibližně odpovídá třem dílkům stavebnice, a tak nebyl problém vytvořit několik typů aktuátorů, které se opakují vedle sebe s rozestupy jednoho dílku.

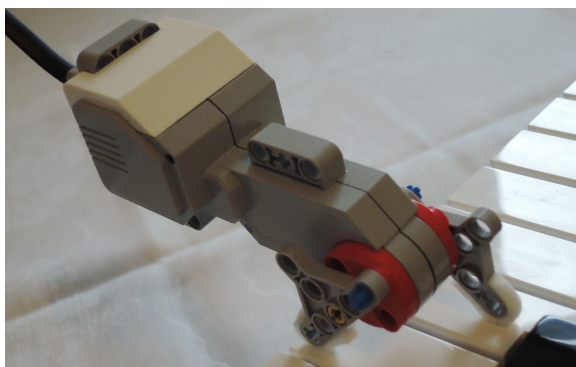
Každá řídicí jednotka má pouze čtyři porty pro ovládání servomotorů, bylo tedy nutné minimalizovat počet potřebných motorů. Jednodušší způsob by byl využívat pouze některé tóny [6, 17]. Takový robot by však zvládl zahrát jen velmi omezené množství skladeb. Proto bylo rozhodnuto, že bude robot ovládat všechny klávesy oktávy, a to s využitím jednoho motoru k ovládání dvou kláves [17, 18]. Otočení motoru jedním směrem stiskne jednu klávesu, otočení motoru opačným směrem stiskne klávesu sousední. Díky tomu postačí k ovládání celé oktávy pouze šest motorů a 1.5 řídicí jednotky.

Kvůli dorovnání rozdílu roztečí stavebnice a kláves, a také kvůli využití celočíselného počtu řídicích jednotek byl robot rozdělen na moduly. Každý modul je schopen plně obsluhovat dvě oktávy klaviatury. Využívá celkem dvanáct motorů a tři řídicí jednotky s plně obsazenými porty motorů.



Obrázek 2.3: Modul robota, pohled shora

Z důvodu rozdílné stavby kláves byly vytvořeny celkem tři základní typy aktuátorů. Tyto typy se postupně opakují a tvoří dohromady modul robota. První typ aktuátoru využívá Velký motor Mindstorms ke stisku dvou sousedních bílých kláves. Rotační pohyb kolmý k ose motoru je převeden pákou na stisk klávesy. Tento typ je v modulu využit pro ovládání dvojic tónů C a D, E a F, G a A.



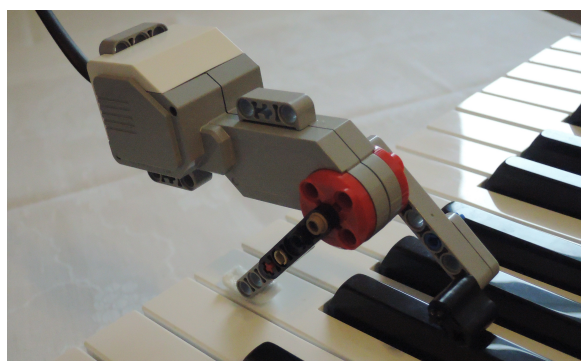
Obrázek 2.4: První typ aktuátoru

Druhý typ aktuátoru využívá Střední motor Mindstorms ke stisku dvou sousedních černých kláves. Rotační pohyb rovnoběžný s osou motoru je převeděn na lineární posuv, který následně stiskne klávesu. Tento typ aktuátoru se v modulu využívá k ovládání dvojic tónů Cis a Dis, Fis a Gis.



Obrázek 2.5: Druhý typ aktuátoru

Jelikož má pianová oktáva lichý počet bílých i černých kláves a klávesy jsou ovládány po dvojicích, byl vytvořen třetí typ aktuátoru, který se podobá prvnímu typu a slouží k ovládnutí zbylých tónů. Využívá také Velký motor Mindstorms a pákový převod ke stisku kláves odpovídajících tónům Ais a H.

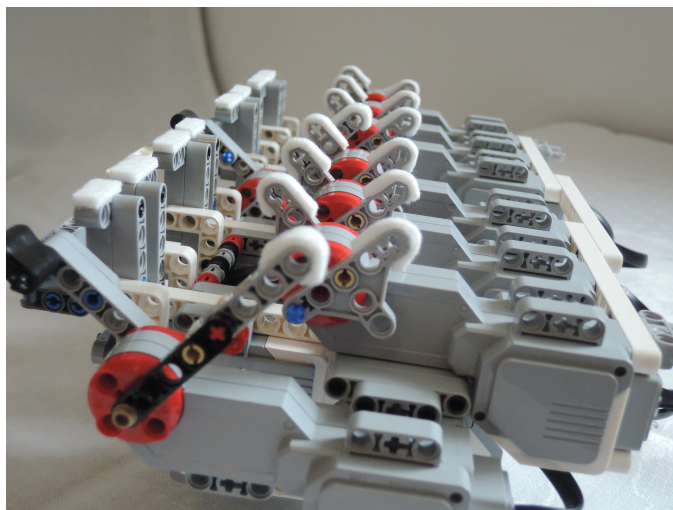


Obrázek 2.6: Třetí typ aktuátoru

Třetí typ aktuátoru by pro stavbu modulu nebyl nutný, jelikož je modul určen pro dvě oktávy, celkový počet bílých i černých kláves je vždy sudý. Vyrábí se však piana s různým počtem oktáv, například piano zde použité má pět oktáv, tedy lichý počet. Proto byl modul robota navržen ke snadnému rozdělení na dvě části. Robot je tak univerzální pro kterékoli piano se standardní roztečí kláves.

Ke stisku kláves je třeba jistá síla, různá pro každé piano. K realizování této síly je využita hmotnost modulu robota. Modul doslova leží na klaviatuře, takže se v klidovém stavu opírá o všechny klávesy. Jakmile nastane čas stisku klávesy, příslušný motor se otočí o určitý počet stupňů, definovaný pro každý typ aktuátoru, a odpovídající klávesa je částí tíhy robota stlačena do požadované pozice. Aby nedošlo ke stisku všech kláves v klidovém stavu, je hmotnost modulu rozložena mezi klávesy a stolec určený k přesnému umístění modulu.

Plastové díly dotýkající se klaviatury byly opatřeny měkkou filcovou vrstvou v místě kolize. Toto opatření jednak zabraňuje odírání klaviatury, omezuje vydávání rušivých klapavých zvuků, a také redukuje nepřesnost hloubky stisku klávesy.

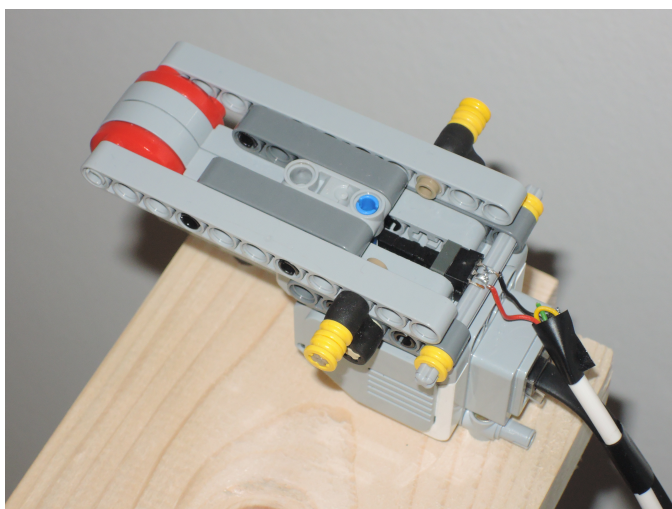


Obrázek 2.7: Otočený modul robota, pohled na aktuátory

2.2.2 Sustain controller

Většina skladeb využívá sustain pedál k prodloužení zahraných tónů, aniž by musely být stisknuty příslušné klávesy. Díky tomu se melodie skladby prolíná s odpovídajícími akordy. Nicméně pedál nemůže být stisknutý po celou dobu hry, musí být v jistých momentech puštěn a znovu stisknut. V opačném případě by se prolínaly i nesouvisející akordy a docházelo by k disharmonii. Snadnější variantou je pedál nepoužívat vůbec, což je možné, ale projeví se to na kvalitě zahrané skladby. Proto byla sestavena část robota pojmenovaná Sustain controller sloužící k ovládní tohoto pedálu. Základem konstrukce je Velký motor Mindstorms, jehož otáčivý pohyb je převáděn na lineární posuv. Pomocí toho je ovládán mikrospínač připojený do kláves kabelem s konektorem Jack 6.3 mm.

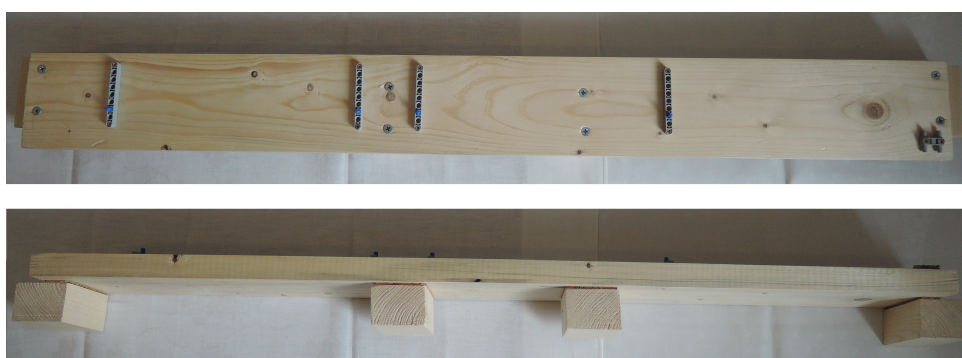
K ovládní tohoto motoru je vyhrazena další řídicí jednotka. Tato jednotka je definována jako hlavní, takže s ní lze ovládat přehrávání skladeb. Inteligentní kostka není nikde přichycena, lze ji tedy snadno uchopit do ruky a ovládat celého robota. Jediným omezením je kabel připojující Sustain controller. V případě úspory řídicích jednotek není nutné používat k ovládní Sustain controlleru samostatnou inteligentní kostku a lze jej připojit ke kterémukoli modulu robota, avšak na úkor některých tónů.



Obrázek 2.8: Sustain controller přichycený ke stolku

2.2.3 Stolek

Ke správnému stisku kláves je zapotřebí, aby byly moduly robota podpírány ve správné výšce. Za tímto účelem byl sestaven dřevěný stolek vyrobený z prkna a čtyř podpěrných kvádrů. Drobná výšková nepřesnost byla upravena vrstvou korku. Při stisku kláves docházelo k pohybu celého modulu po klaviatuře. Proto byly ke stolku připevněny díly stavebnice Lego tak, aby bylo možné přesně přichytit moduly v požadovaných pozicích. V pravé části stolku je také umístěn díl pro připevnění Sustain controlleru. Přesné umístění stolku u piano zajišťují podpěrné kvádry. Krajní dvojice stabilizuje posun stolku ve směrech vpravo, vlevo a prostřední dvojice se opírá o piano zepředu. Díky tomu jsou moduly vždy na požadovaných pozicích.



Obrázek 2.9: Stolek pro robota; pohled shora (nahore), pohled zepředu (dole)

Kapitola 3

Programy

První část kapitoly je věnována programovacímu jazyku a programům využitým při tvorbě řídicího programu robota a konverzního programu vstupních dat popisovaných v druhé části kapitoly.

3.1 Použitý software

Vzhledem ke komplexnosti úlohy byl zvolen objektový návrh programu. Proto byly oba vytvořené programy napsány v jazyce Java. Aby nebylo nutné přepisovat skladby do digitálního formátu podporovaného řídicí jednotkou ručně, byl pro import skladeb využit formát digitální hudby MIDI.

3.1.1 Java

Počátkem devadesátých let dvacátého století vyvinul tým programátorů firmy Sun Microsystems programovací jazyk Oak, který měl sloužit k řízení domácích spotřebičů [19]. Později byl jazyk přejmenován podle amerického slangového označení kávy – Java. V roce 1995 byl jazyk oficiálně představen veřejnosti, která ho díky jeho vlastnostem rychle přijala, a Java se tak stala jedním z nejpoužívanějších programovacích jazyků. Sun Microsystems v roce 2007 zveřejnil zdrojové kódy Javy a jazyk je nadále vyvíjen veřejností.

Syntaxe Javy je velmi podobná jazykům C a C++, eliminuje však některé nízkourovňové prvky (např.: ukazatelé, příkaz goto) [19]. Jazyk je objektově orientovaný, všechny datové typy, kromě primitivních, jsou objekty (třídy). Jednou z hlavních výhod Javy je, že je interpretovaná. Zdrojový kód programu tedy není překládán přímo do strojového kódu konkrétní architektury procesoru, ale do tzv. bajtkódu (Java bytecode), který je spouštěn interpretem Javy – tzv. virtuálním strojem Javy (Java Virtual Machine) [19]. Díky tomu je možné spustit zkompilovaný program na kterémkoliv počítači/mikropočítači, který má k dispozici virtuální stroj Javy, nebo dokáže vykonávat přímo bajtkód.

■ 3.1.2 LeJOS EV3

K programování stavebnic Mindstorms je oficiálně určený grafický programovací jazyk, kdy je program tvořen spojováním funkčních bloků. Mimo to existují různé verze firmwaru (řídícího programu inteligentní kostky) vyvíjené dobrovolníky, které umožňují programovat inteligentní kostku v různých jazycích. Mezi ně patří i projekt LeJOS, který nabízí virtuální stroj Javy pro jednotky RCX, NXT i EV3 rozšířený o třídy ovládající hardware specifický pro stavebnici Mindstorms (např. ovládání motorů, čtení senzorů). Verze LeJOS EV3, na rozdíl od předchozích, podporuje celé jádro knihoven Java, nicméně stále není dokončena a její spouštění probíhá z paměťové karty. Nelze ji nahrát do interní paměti kostky.

■ 3.1.3 MIDI

S rozvojem elektrotechniky došlo v druhé polovině dvacátého století i k rozvoji elektronických hudebních nástrojů. Počátkem osmdesátých let se již začínaly objevovat první pokusy o digitální přenos hudby, avšak každý výrobce hudebních nástrojů měl vlastní komunikační protokol, a nástroje tak nebyly mezi sebou kompatibilní. Tento problém si brzy uvědomil zakladatel firmy Roland a spolu s dalšími výrobci hudebních nástrojů (Oberheim electronics, Sequential circuits, Yamaha, Korg, Kawai) [20] navrhli standard MIDI (Musical Instrument Digital Interface – Digitální rozhraní hudebních nástrojů), který specifikuje komunikační protokol i fyzickou vrstvu realizovanou sériovou proudovou smyčkou.

Standard zachycuje hudbu jako sled událostí (např. stisk nebo uvolnění klávesy piana) v různých kanálech. Každý kanál představuje jeden nástroj,

maximální počet kanálů v MIDI je 16 [20]. Události mají svůj typ a parametry: nejprve je specifikován typ události jednobytovým číslem (Status byte) a po něm může následovat jeden či více parametrů, každý reprezentovaný jednobytovým číslem (Data Byte). Například stisk klávesy C4 v kanálu 0 je uveden Status bytem 144 (decimálně), následuje datový byte 60 označující tón C4 a dalším datovým bytem je uvedena rychlost, tedy dynamika stisku [21]. Stisk klávesy je stejně jako třeba změna nástroje tzv. kanálová událost, tedy je vždy pouze pro jeden kanál. Druhým typem událostí jsou systémové události, například změna tempa nebo rytmu, které platí pro všechny kanály [20].

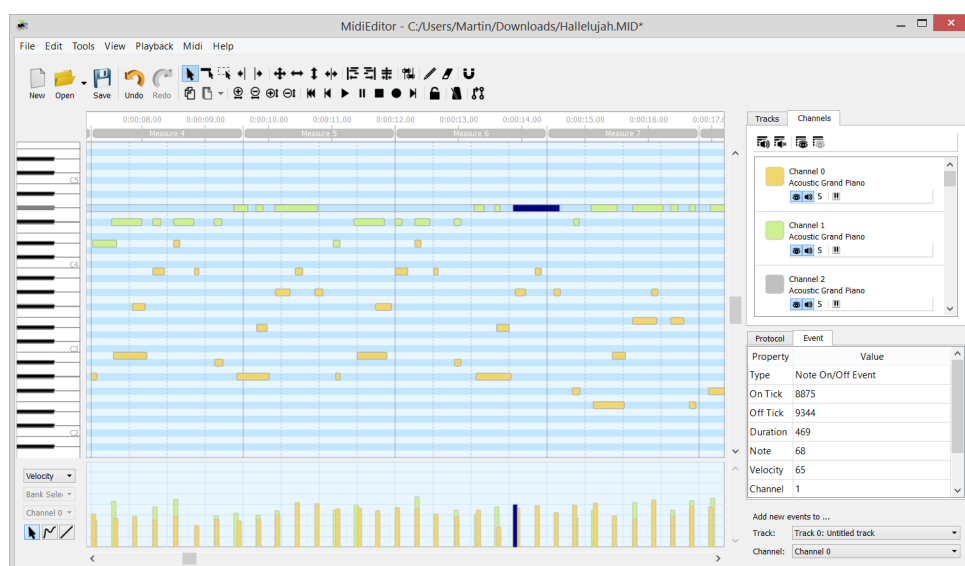
Kromě živého přenosu lze MIDI zapsat do souboru, a to buď záznamem hry na MIDI kompatibilní nástroje, nebo zápisem not na počítači pomocí softwaru k tomu určenému. Každá událost má navíc čas, kdy má být vykonána. V jednom souboru může být více stop (Tracků), kdy každá stopa obsahuje maximálně 16 kanálů. V práci bylo využito MIDI souborů ke zjednodušení tvorby vstupních dat, podle kterých robot ovládá klávesy.

■ 3.1.4 MidiEditor

Soubory MIDI většinou nestačí pouze stáhnout z internetu, ale je třeba je upravit (např. překrývající se sousední noty, chybějící sustain pedál). Pro tento účel byl při práci používán volně dostupný program MidiEditor [3]. Tento program vizualizuje MIDI události ve dvourozměrném prostoru, kde první rozměr je čas a druhý rozměr je tón nebo skupina systémových událostí. Různá barva událostí rozlišuje kanály, do kterých náleží. Události je možné přidávat, odebírat nebo upravovat jejich typ i parametry. Lze také ztlumit vybrané kanály nebo celé stopy, popřípadě měnit hudební nástroje kanálů. Program využívá systémový syntetizér MIDI, díky čemuž je schopný přehrávat vybranou část nebo celý soubor.

■ 3.2 Vlastní software

Programové vybavení robota je rozděleno na dvě části – řídicí program robota kompilovaný pro prostředí LeJOS EV3 a program pro konverzi vstupních dat kompilovaný pro osobní počítač. Požadovaná skladba k přehrávání musí být nejprve vytvořena nebo stažena z internetu ve formátu MIDI. Z toho je následně převedena pomocí konverzního programu na soubor formátu



Obrázek 3.1: Grafické uživatelské rozhraní programu MidiEditor [3]

EV3song, což je formát vstupních dat řídicího programu. Takový soubor je nahrán do hlavní řídicí jednotky, ve které může být načten řídicím programem, a konečně přehrán robotem.

3.2.1 Řídicí program robota - PianoPlay

Jelikož robot obsahuje několik řídicích jednotek, bylo nutné jejich práci synchronizovat. K tomu byl zvolen model Master/Slave, kde vybraná jednotka vydává příkazy a ostatní jednotky ji poslouchají. Řídicí program nazvaný PianoPlay je pro oba typy jednotek stejný, jeho funkce se liší až po spuštění a vybraní postavení jednotky uživatelem.

Ke komunikaci mezi jednotkami je využita síť PAN (Personal Area Network) [22], což je implementace propojení více inteligentních kostek v prostředí LeJOS EV3. Síť PAN podporuje spojení mezi kostkami pomocí USB, Bluetooth a Wi-Fi. V bakalářské práci bylo použito spojení Bluetooth, jelikož nevyžaduje žádný hardware navíc oproti ostatním možnostem. Nevýhodou tohoto spojení je nižší rychlost komunikace, než u ostatních typů [23], nicméně pro účely této práce je rychlost dostačující. Komunikace v síti probíhá pomocí protokolu TCP/IP. Vybraná jednotka (v této práci je to inteligentní kostka ovládající Sustain controller) je nastavena do módu Access Point s IP adresou 10.0.1.1. Ostatní jednotky jsou nastaveny do módu BT Client. Po spuštění vyhledají přístupový bod a ten jim přidělí unikátní IP adresu.

Program je rozdělen do tří balíčků tříd – *notes*, *pianoplayev3* a *utils*. Balíček *utils* obsahuje pouze třídu *Buttons*, která byla vytvořena kvůli nedostatkům původní třídy *Button* systému LeJOS. Konkrétně se jedná o doplnění chybějící metody *waitForRelease()*, která je využívána při každé obsluze stisku některého z tlačítek řídicí jednotky.

V balíčku *notes* jsou umístěny třídy *Note* a *Song*. Tyto třídy slouží k reprezentaci hudby a obsahují všechny potřebné metody k práci s ní.

■ Třída Note

Tato třída slouží pouze k uchování informace o události tónu. Obsahuje specifikaci, o který tón se jedná, v jaký čas událost nastane a jestli má být přehrávání tónu zahájeno, nebo ukončeno. Jediná metoda této třídy slouží k porovnání tónu události a specifikovaného tónu. Třída implementuje rozhraní *Serializable*, čehož je využíváno při ukládání/načítání skladeb a jejich odesílání po síti.

■ Třída Song

K reprezentaci skladby slouží třída *Song*. K tomu využívá seznam obsahující jednotlivé události ve formě objektů *Note*. Kromě tohoto seznamu uchovává třída ještě název skladby. Instanci třídy lze vytvořit dvěma způsoby – načtením událostí ze souboru formátu EV3song, nebo vytvořením prázdné skladby se specifikovaným názvem. Při využití druhého způsobu jsou události do skladby přidávány později pomocí metody k tomu určené. Následně je zapotřebí seřadit události v seznamu podle jejich časů vykonání k usnadnění pozdějšího přehrávání. Tuto činnost obsluhuje metoda využívající algoritmus Bubble sort [24, 25]. Princip algoritmu spočívá v opakovaném porovnávání sousedních dvojic čísel. Největší číslo vždy “probublá“ na konec řady. Tento algoritmus je časově poměrně náročný, ale vzhledem k počtu událostí skladeb ho lze použít.

Další metody, které třída obsahuje, slouží k přístupu k jednotlivým událostem, k jejich odstranění, nebo ke zjištění jejich počtu. Poslední metoda ukládá události skladby do souboru formátu EV3song. Formát byl vytvořen kvůli přenosu dat z konverzního programu v počítači. Struktura souboru je jednoduchá: nejprve je zapsán počet událostí a za ním následují serializované objekty *Note*. Název souboru odpovídá názvu skladby.

Balíček *pianoplayev3* obsahuje třídy *MainMenu*, *Player*, *Master*, *Slave* a *PianoPlayEV3*, které řídí celkový chod robota. Poslední jmenovaná třída obsahuje hlavní metodu *main()*, která je spuštěna po startu programu. Jediný úkol této metody je vytvořit instanci třídy *MainMenu*.

■ Třída *MainMenu*

Po spuštění programu se uživateli na LCD displeji zobrazí menu s výběrem postavení řídicí jednotky. Za toto menu zodpovídá třída *MainMenu*. Displej je rozdělen na řádky. Některé řádky jsou považovány za položky menu. Na těchto řádcích jsou vykreslena jména položek (*Master*, *Slave*) a kurzor. Třída obsahuje proměnnou uchovávající řádek pozice kurzoru. Stiskem tlačítek *Up*, *Down* uživatel mění pozici kurzoru, tlačítkem *Enter* vybere položku z menu nebo tlačítkem *Escape* ukončí program. Stejný způsob grafického uživatelského rozhraní je využit i v následujících třídách. Po provedení výběru postavení řídicí jednotky je vytvořena instance příslušné třídy *Master* nebo *Slave*.

■ Třída *Player*

Přehrávání skladeb je v kompetenci třídy *Player*. Tato třída má dva režimy funkce. Po vytvoření instance se nachází v režimu nastavení aktuátorů. Na displeji je zobrazeno schéma definující, který port motoru ovládá kterou klávesu piana jakým typem aktuátoru. Toto nastavení se při každé změně uloží do konfiguračního souboru a po restartu programu se opět nahraje. Uživatel tlačítka *Up*, *Down* vybírá port motoru, tlačítka *Left*, *Right* vybírá sloupec upravující tón nebo typ aktuátoru a tlačítkem *Enter* mění vybraný parametr. Tóny zobrazené na displeji odpovídají levým tónům z dvojice ovládané aktuátorem (při standardním pohledu na klávesy).

Po nastavení aktuátorů může uživatel přejít do režimu přehrávání. Při přechodu do tohoto režimu dojde k inicializaci motorů – podle nastavených aktuátorů se vytvoří instance tříd ovládajících motory (Velký motor *Lego*, Střední motor *Lego*) na příslušných portech (porty *A*, *B*, *C*, *D*) a nastaví se jim maximální rychlost a zrychlení. Podle zadaných tónů a typů aktuátorů jsou dopočítány jejich doplňky. Všechny tyto tóny jsou uloženy, aby program později rozpoznal, které tóny je schopen přehrát. Poté začne přehrávání skladby, která byla nastavena nadřazenou třídou (třída *Master* nebo *Slave*) pomocí objektu *Song*. Nadřazená třída vždy zavolá metodu *update()* s parametrem aktuálního času. Z přehrávané skladby je přečtena první událost. Pokud je

její čas provedení větší než aktuální čas, nic se nestane. V opačném případě dojde k porovnání tónů. Pokud je daná jednotka nastavena na přehrávání tónu této události, zavolá metodu pro provedení události. Ta otočí příslušným motorem o určitý počet stupňů odpovídající typu aktuátoru. Vhodný počet stupňů byl určen experimentálně pro každý typ aktuátoru. Poté je událost odstraněna ze seznamu událostí skladby. Tlačítkem Escape může uživatel pozastavit přehrávání a navrátit se tak do režimu nastavování aktuátorů.

■ Třída Master

Tato třída je určena pro ovládání celého robota a její použití je očekáváno na řídicí jednotce v módu Access point s IP adresou 10.0.1.1. Po vytvoření instance poslouchá na portu 2222 a umožňuje tak podřízeným jednotkám se připojit. Pomocí displeje vybízí uživatele k zapnutí podřízených jednotek. Po stisku tlačítka Enter zobrazí počet připojených podřízených jednotek pro kontrolu komunikace. Následně třída najde všechny soubory formátu EV3song v kořenové složce a vypíše jejich názvy na displej v podobě menu. Uživatel může tlačítka Up, Down vybírat v seznamu, nebo tlačítka Left, Right přepnout do módu přehrávání všech skladeb ve smyčce. Po výběru skladby uživatelem je nahrán příslušný soubor do objektu *Song*, který je předán objektu *Player*. Po spuštění přehrávání začne třída počítat čas, kterým opakovaně aktualizuje objekt *Player*.

Pro ovládání podřízených jednotek využívá třída čtyři typy příkazů: Play, Stop, Sync time a Receive song. První dva typy slouží ke spuštění nebo zastavení přehrávání a nevyžadují žádné další parametry. Třetí typ je určen k synchronizaci času mezi jednotkami, aby přehrávaly stejnou část skladby. Parametrem tohoto příkazu je aktuální čas přehrávání. Poslední příkaz odesílá podřízeným jednotkám skladbu zvolenou k přehrávání. Prvním parametrem příkazu je počet událostí skladby, aby podřízené jednotky věděly, kolik událostí mají přijmout. Následuje odeslání názvu skladby. Nakonec jsou postupně odeslány všechny události skladby v podobě serializovaných objektů *Note*.

■ Třída Slave

Podřízené jednotky jsou spravovány třídou *Slave*. Po vytvoření instance se třída pokusí připojit k nadřazené jednotce s adresou 10.0.1.1 a portem 2222. V případě neúspěšného připojení informuje uživatele a ukončí program. V opačném případě vytvoří instanci třídy *Player* a očekává příkazy od nadřazené

jednotky. Při přijetí příkazu Play začne třída počítat čas a aktualizuje s ním objekt *Player*, při přijetí příkazu Stop přestane počítat čas, při přijetí příkazu Sync time aktualizuje třída čas podle přijatých dat a při přijetí příkazu Receive song vytvoří novou instanci třídy *Song* s názvem z přijatých dat a postupně přidává přijaté události. Události objektu *Song* následně pro jistotu setřídí podle času vykonání a předá objektu *Player*.

```

PianoPlay EV3
> Slave
  Master

< Select a song: >
> Gaudeamus Igitur
  Hallelujah.EV3song
  Bach Ave Maria.E
  Your Reality.EV3
  Amelie.EV3song
  test song.EV3son

Hallelujah
> Play song
A: c 4 1
B: e 4 1
C: g 4 1
D: a # 4 3
S

```

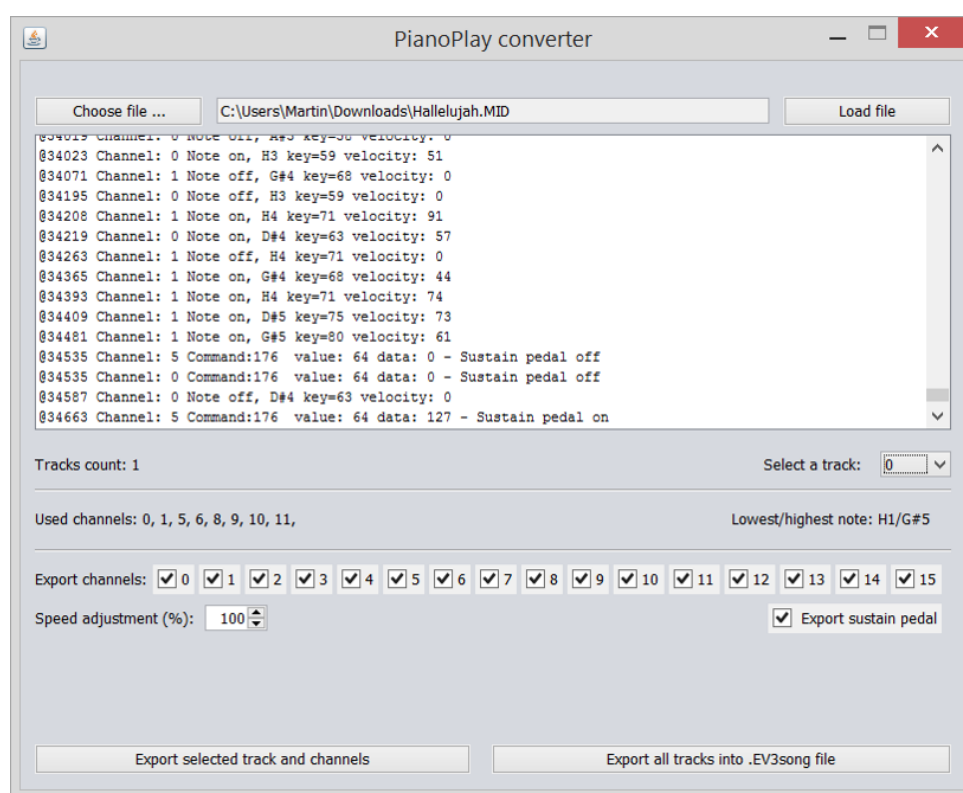
Obrázek 3.2: Uživatelské rozhraní robota - Výběr postavení třídy *MainMenu* (vlevo), výběr skladby třídy *Master* (uprostřed), schéma ovládaných tónů třídy *Player* (vpravo)

Při komunikaci je využíván pouze jeden směr Master -> Slave, časová synchronizace je tak bez zpětné vazby a čas hlavní kostky se od času podřízené kostky liší o rozdíl mezi odesláním a přijetím příkazu. Proto je v instanci třídy *Player* vytvořené třídou *Master* statické zpoždění přehrávání o 90 ms určené experimentálně. Dalším zmírněním problému je použití hlavní jednotky pouze k ovládní Sustain controlleru, kde je časový rozdíl přehrávání ještě méně rozpoznatelný.

3.2.2 Konverzní program pro PC - PianoPlay converter

Pro import skladeb z MIDI byla vytvořena aplikace pro osobní počítač nazvaná PianoPlay converter. Aplikace využívá knihovnu Swing pro tvorbu grafického uživatelského rozhraní. Uživatel nejprve specifikuje cestu k MIDI souboru, který chce konvertovat. Po stisku tlačítka Load file se v textovém poli zobrazí podrobný výpis všech událostí zvoleného souboru. Pro čtení MIDI souborů byla využita upravená verze kódu z tohoto webu [26]. Pod výpisem je zobrazen počet stop souboru. Uživatel může vybrat jednu ze stop a pro tu jsou zobrazeny informace o využitých kanálech a informace o nejnižším a nejvyšším tónu skladby. V případě potřeby je možné zpomalit nebo zrychlit přehrávání skladby a také lze povolit nebo zakázat export ovládní sustain pedálu. Tento pedál je ve formátu EV3song reprezentován jako tón C -2 přehrávaný aktuátorem prvního typu (odpovídající tón je příliš nízký, a proto není ve skladbách využíván).

Následně má uživatel dvě možnosti: exportovat všechny stopy do souboru EV3song, nebo exportovat pouze vybranou stopu s vybranými kanály. Pro reprezentaci hudby využívá program stejný balíček *notes*, jako předchozí program. Při exportu je tedy vytvořena instance třídy *Song* s názvem skladby odpovídajícím názvu MIDI souboru. Poté jsou postupně přečteny všechny události souboru a porovnány se zvolenými požadavky na export. Vyhovující události jsou spolu s dopočítaným časem provedení vloženy do seznamu objektu *Song*. Po vložení všech událostí je zavolána metoda třídy *Song* k seřazení událostí podle času vykonání. Na závěr je využita další metoda této třídy, a to uložení seznamu událostí do souboru formátu EV3song.



Obrázek 3.3: Grafické uživatelské rozhraní programu PianoPlay converter

Kapitola 4

Webová stránka

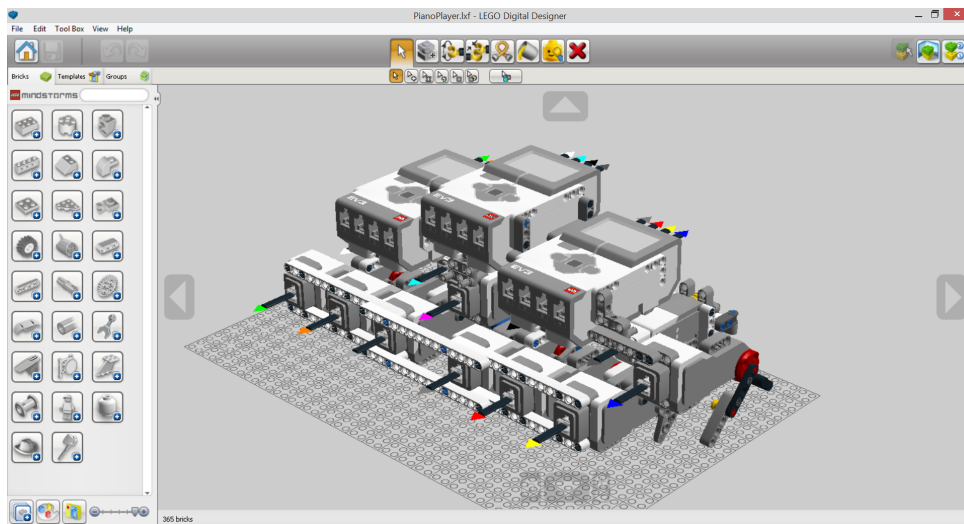
V rámci této práce byla vytvořena webová stránka dokumentující robota, která je určená veřejnosti. Stránka obsahuje stručný popis použitých mechanismů, popis funkce programů a návod na instalaci a obsluhu softwaru. Dále jsou zde ke stažení umístěny spustitelné soubory obou vytvořených programů, ukázkové skladby ve formátu EV3song a pdf soubory s návody na stavbu robota vytvořené pomocí modelovacího programu Lego Digital Designer.

Webová stránka bude umístěna na internetový portál Robosoutěže [8], aby byla dostupná pro každého, kdo by chtěl robota replikovat. Na tomto portálu se již nachází několik stránek s návody na stavbu Lego robotů s různými účely. Proto byla využita jedna z nich (Skladač Rubikovy kostky [27]) jako šablona, kvůli zachování stylu portálu. Stránka je psána v jazyce HTML s využitím kaskádových stylů.

4.1 Lego Digital Designer

Stavebnice Lego existuje i v digitální verzi. Tou je volně dostupný modelovací program Lego Digital Designer vytvořený a distribuovaný společností Lego [4]. Hlavní částí tohoto programu je 3D editor, tedy zobrazovaný prostor, ve kterém se lze pohybovat. Do tohoto prostoru uživatel umísťuje virtuální Lego díly, které vybírá v postranní záložce. V horní liště programu se nachází nástroje pro výběr jednoho dílu nebo jejich skupiny a nástroje pro práci s ní. Uživatel tak může sestavit z Lega cokoli, co si vymyslí, bez omezení počtem

dílků stavebnice, které vlastní. V pravém horním rohu aplikace se nachází ikona přepnutí do módu stavebních instrukcí. Program vygeneruje návod na stavbu vytvořené konstrukce spolu se seznamem použitých dílů. Tato funkce byla využita při tvorbě návodu na stavbu robota pro webovou stránku.



Obrázek 4.1: Grafické uživatelské rozhraní programu Lego Digital Designer [4]



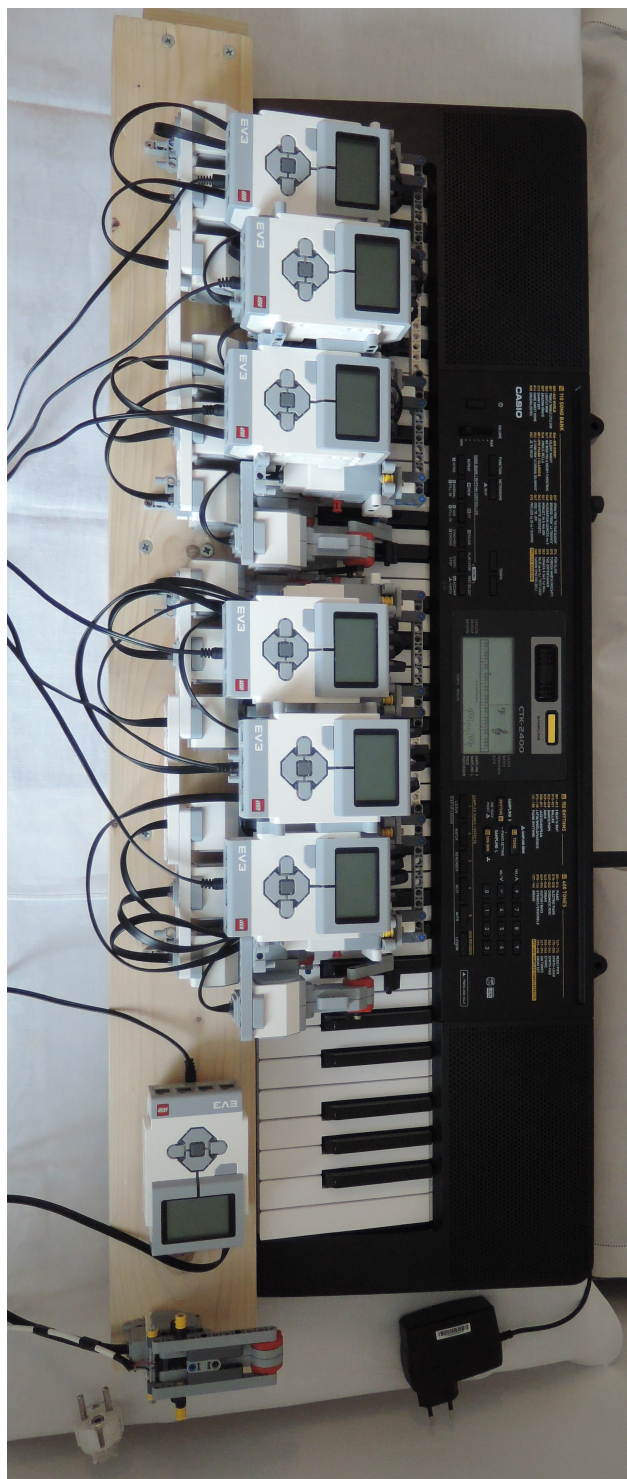
Kapitola 5

Závěr

V rámci práce byla navržena konstrukce robota hrajícího na piano s potřebným příslušenstvím. Tento návrh byl realizován za použití stavebnice Lego Mindstorms EV3. Sestavený robot dokáže plně ovládat čtyři oktávy pianové klaviatury včetně sustain pedálu. Pro vytvořený hardware byl napsán řídicí program koordinující práci několika inteligentních kostek. Vstupní data řídicího programu jsou vytvářena konverzním programem, který je další částí této práce. Program načítá hudbu z digitálního hudebního formátu MIDI a umožňuje uživateli exportovat zvolenou část skladby.

Díky množství aktuátorů dokáže robot zahrát některé melodie (například složité akordy), na které dvojice lidských rukou nestačí. Výhodou a nevýhodou zároveň je využití jednoho motoru ke stisku dvou kláves. Tento způsob sice ušetří polovinu potřebných motorů, ale omezuje způsob hry na piano. Proto robot nezvládne rychle po sobě opakovat tóny ovládané jedním motorem. Robot také nezvládne rychle po sobě opakovat ten samý tón. To je ale způsobeno limity samotných motorů.

Robot je tedy vhodný k přehrávání skladeb v mírnějším tempu nebo pro doprovod dalších hudebních nástrojů. K reprezentačním účelům je robot vybaven funkcí přehrávání všech skladeb ve smyčce, což bude jistě užitečné pro vystavení. Případní zájemci mohou využít webovou stránku vytvořenou v rámci této práce k porozumění principu funkce robota nebo i k jeho stavbě.



Obrázek 5.1: Finální verze robota se stolkem a elektrickými klávesami

Příloha A

Literatura

- [1] CASIO AMERICA INC. Casio CTK-2400, 2018. [Online] <https://www.casio.com/products/electronic-musical-instruments/portable-keyboards/ctk-2400>.
- [2] Laurens Valk. Ev3 and nxt: Difference and compatibility. ROBOT-SQUARE, 16. 7. 2013. [Online] <http://robotsquare.com/2013/07/16/ev3-nxt-compatibility/>.
- [3] Markus Schwenk. Midieditor - graphical interface to edit, play, and record midi data. SourceForge, 2018. [Online] <http://midieditor.sourceforge.net/>.
- [4] The Lego Group. Lego digital designer. The Lego Group, 2018. [Online] <https://www.lego.com/en-us/ldd>.
- [5] Příspěvatelé Wikipedie. Player piano. Wikipedia, The Free Encyclopedia, 7. 3. 2018. [Online] https://en.wikipedia.org/wiki/Player_piano.
- [6] TECHNICally Possible. On longing piano cover by lego mindstorms ev3. YouTube, 28. 4. 2015. [Online] <https://www.youtube.com/watch?v=TlmEpdYKwpI>.
- [7] Keane G Reagan, Carson. Lego mindstorm piano player. YouTube, 15. 7. 2015. [Online] https://www.youtube.com/watch?v=oX0jtxq5_4A.
- [8] Katedra řídicí techniky. Robosoutěž. ČVUT v Praze, 2018. [Online] <https://robosoutez.fel.cvut.cz/>.
- [9] Příspěvatelé Wikipedie. Oktáva (hudba). Wikipedie: Otevřená encyklopedie, 4. 10. 2017. [Online] [https://cs.wikipedia.org/wiki/Okt%C3%A1va_\(hudba\)](https://cs.wikipedia.org/wiki/Okt%C3%A1va_(hudba)).



Příloha B

Obsah přiloženého CD

- Složka models – modely robota pro program Lego Digital Designer
- Složka programs – programy PianoPlay.jar, PianoPlayConverter.jar
- Složka sources – zdrojové kódy programů
- Složka songs – ukázkové skladby ve formátech MIDI a EV3song
 - J. S. Bach, C. Gounod – Ave Maria
 - Leonard Cohen – Hallelujah
- Složka videos – videa robota v akci
- Složka website – webová stránka s návodem na stavbu a obsluhu robota
- Soubor thesis.pdf - tato práce