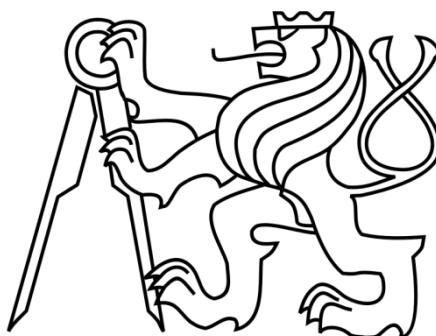


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

ÚSTAV MECHANIKY, BIOMECHANIKY A MECHATRONIKY

Odbor mechaniky a mechatroniky



Bakalářská práce

Úprava výukového robota pro ovládání z prostředí

Matlab/Simulink

Praha, 2016

Ondřej Kolda

Anotační list

Jméno autora:	Ondřej Kolda
Název bakalářské práce:	Úprava výukového robota pro řízení z prostředí Matlab/Simulink
Anglický název:	Modification of the teach-robot for control using Matlab/Simulink
Akademický rok:	2015/2016
Obor studia:	Teoretický základ strojního inženýrství
Ústav/odbor:	Ústav mechaniky, biomechaniky a mechatroniky Odbor Mechaniky a mechatroniky
Vedoucí bakalářské práce:	Ing. Petr Beneš, Ph.D.
Bibliografické údaje:	Počet stran: 35 Počet obrázků: 41 Počet příloh: 1 CD
Klíčová slova:	Teach-robot, Arduino, Motor-Shield, Hallova sonda, dopředná kinematika, inverzní kinematika, Matlab, Simulink, řízení
Keywords:	Teach-robot, Arduino, Motor-Shield, Hall effect sensor, kinematics, inverse kinematics, Matlab, Simulink, control
Anotace:	Předmětem práce je robotický manipulátor Teach-robot. Jsou vytvořeny výpočtové modely pro řešení dopředné a inverzní kinematické úlohy. Je sestavena nová řídicí elektronika. Práce se zabývá vytvořením řídicího programu v prostředí Matlab/Simulink
Abstract:	The object of the study is a robotic manipulator Teach-robot. Computational models are created for solving forward and inverse kinematic problems. A new control electronics is assembled. The study deals with a creation of a new control program in Matlab/Simulink software.

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

V Praze, dne

.....

Podpis

Poděkování

Tímto bych rád poděkoval vedoucímu bakalářské práce, panu Ing. Petru Benešovi, PhD., za odborné vedení práce, ochotu při poskytování rad i připomínek a především za psychickou podporu v těžkých chvílích, bez které by tato práce nebyla nikdy dokončena.

Obsah

Anotační list	3
Prohlášení	5
Seznam obrázků	7
1.... Úvod.....	8
2.... Klasifikace robotů	9
3.... Arduino.....	12
3.1. Arduino Mega 2560	12
3.2. Motor Shield	13
3.3. Simulink support package for arduino hardware.....	13
3.3.1. Instalace knihovny Arduino	13
3.3.2. Nahrání programu do Arduina.....	15
4.... Teach-robot	17
4.1. Konstrukční provedení	17
4.2. Provedení elektronických obvodů	18
4.2.1. Zapojení motorů.....	18
4.2.2. Hallova sonda	19
5.... Kinematický model Teach-robotu.....	21
5.1. Dopředná kinematika	21
5.2. Inverzní kinematika	25
6.... Řídicí program.....	29
6.1. Ovládání motoru	29
6.1.1. Skutečná poloha motoru.....	29
6.1.2. Nastavení rychlosti motoru	30
6.1.3. Nastavení směru otáčení	31
6.2. Řízení Teach-robotu.....	32
7.... Závěr	34
8.... Literatura.....	35

Seznam obrázků

Obr. 1 - Automatizovaná výroba automobilů.....	7
Obr. 2 - Sériový robot fy Yaskawa.....	9
Obr. 3 - Paralelní robot fy Fanuc.....	9
Obr. 4 - Hybridní robot.....	9
Obr. 5 - Servisní průzkumný a pyrotechnický robot.....	9
Obr. 6 - Rozdělení manipulačních zařízení.....	10
Obr. 7 - Arduino Mega2560	11
Obr. 8 - Motor Shield.....	12
Obr. 9 - Instalace knihovny Arduino	13
Obr. 10 - Support Package Installer	13
Obr. 11 - Výběr knihoven.....	13
Obr. 12 - Knihovna Arduino.....	14
Obr. 13 - Ovládání diody	15
Obr. 14 - Configuration Parameters	15
Obr. 15 - Spuštění programu.....	15
Obr. 16 - Teach-robot ovládací panel	17
Obr. 17 - Radiální motor u podstavy robotu.....	17
Obr. 18 - Uchycení lineárních pohonů.....	17
Obr. 19 - Elektronický obvod	17
Obr. 20 - Načítání Hallovy sondy	18
Obr. 21 - Načítání Hallovy sondy, osa x - čas v sekundách, osa y - signál sondy.....	19
Obr. 22 - Ošetřený výstupní signál sondy, osa x - čas v sekundách, osa y - signál sondy.....	19
Obr. 23 - Schéma Teach-robotu.....	20
Obr. 24 - Základní pohyby	21
Obr. 25 - Transformace souřadnicových systémů.....	22
Obr. 26 - Minimální dosah YZ.....	23
Obr. 27 - Maximální dosah YZ.....	23
Obr. 28 - Maximální dosah YZ.....	23
Obr. 29 - Minimální dosah YZ.....	23
Obr. 30 - Maximální rozsah 3D	23
Obr. 31 - Minimální rozsah 3D.....	23
Obr. 32 - Natočení roviny robota.....	24
Obr. 33 - Sklon čelistí.....	25
Obr. 34 - Poloha bodu A.....	26
Obr. 35 - Úhel úchopového mechanismu	27
Obr. 36 - Poloha motoru.....	29
Obr. 37 - Spouštění motoru.....	30
Obr. 38 - Směr otáčení.....	30
Obr. 39 - Řídicí program motoru	30
Obr. 40 - Řídicí program Teach-robotu.....	31
Obr. 41 - Propojení Hardware	32

1. Úvod

V současné době je obtížné představit si svět okolo nás a fungování lidské společnosti, tak jak ji známe, bez zásahu robotických systémů. Všichni jsme jimi nějakým způsobem ovlivněni. Někteří z nás přímo jejich užíváním. Usnadňují nám některé neoblíbené práce. Příkladem může být automatický vysavač v domácnostech. V jistých situacích nám umožňují vykonávat nedosažitelnou nebo nebezpečnou práci s dostatečným odstupem, k čemuž nám poslouží pyrotechnické roboty případně vesmírná vozítka sbírající vzorky hornin na Měsíci. Chirurgické roboty pomáhají zachraňovat životy. Robotické protézy vracují některým lidem schopnost uchopit a manipulovat s předměty. Je možné najít mnoho případů, kdy roboty přímo zasahují do našich životů. Nemusíme se však dostat do přímého kontaktu, a přesto narazíme na stopy jejich vlivu. Téměř všichni jsme nepřímě ovlivněni existencí robotů a jejich schopnostmi. Řada běžných věcí, které denně používáme, vděčí za svojí existenci průmyslovým robotům. Ty jsou schopny pracovat s přesností, rychlostí a vytrvalostí jaké i nepreciznější a nejšikovnější člověk nikdy nemůže být schopen. Nejlepším příkladem jsou automobily, které se vyrábějí v automatických robotizovaných linkách, viz obr. 1.



Obr. 1 - Automatizovaná výroba automobilů¹

¹Automatizovaná výroba automobilů. In: *ABB* [online]. © Copyright 2016 ABB [cit. 18.8.2016]. Dostupný z: <http://www.abb.com/cawp/seitp202/f528fea6d4d93db5482579130028c06b.aspx>

V této práci se budu zabývat vytvořením prostředí pro řízení výukového robotu Teach-robot prostřednictvím programu Matlab/Simulink. Robot je vlastnictvím Ústavu mechaniky, biomechaniky a mechatroniky na Fakultě strojní ČVUT v Praze. Je určen pro použití ve výuce. Studentům umožní vyzkoušet si ovládání malého modelu robotu, které se principiálně neliší od ovládání velkých průmyslových robotů.

Cíle práce:

1. Seznámení se s řízením platformy Arduino z prostředí Matlab/Simulink
2. Seznámení se s konstrukcí Teach-robotu
3. Připravení modelů pro výpočet dopředné a inverzní kinematické úlohy
4. Připravení řízení z prostředí Matlab/Simulink

2. Klasifikace robotů

Množství požadavků na potřebné schopnosti a vlastnosti robotů k vykonání rozdílných úkonů dalo vzniknout nesčetnému množství variací robotických systémů, které se od sebe odlišují jak konstrukčně, tak funkčně. Vzniklo tak mnoho odlišných kritérií, které umožňují rozřadit roboty do skupin [1].

Mezi tato kritéria patří z konstrukčního hlediska druh pohonů. Může se jednat buď o dnes nejběžnější způsob pohonu prostřednictvím elektromotorů, nebo hydraulický pohon, který se využívá tehdy, je-li zapotřebí velká síla, eventuálně o pneumatický pohon pro vysoké rychlosti.

Dalším konstrukčním kritériem je kinematická struktura robotu. Ta je dvojího typu. Paralelní nebo sériová, v některých případech je využita i jejich kombinace, která se označuje jako hybridní. Příklady těchto konstrukcí jsou znázorněny na obrázcích 2 až 4.



Obr. 2 - Sériový robot fy Yaskawa²



Obr. 3 - Paralelní robot fy Fanuc²



Obr. 4 - Hybridní robot²

Ke konstrukčním kritériím můžeme také přiřadit počet stupňů volnosti. Zde dělíme roboty do tří skupin. Univerzální roboty se šesti stupni volnosti, redundantní roboty s více jak šesti stupni volnosti a deficitní roboty s méně než šesti stupni volnosti.

Dále lze rozdělit roboty z funkčního hlediska. K těmto kritériím patří oblast nasazení. Jedná se o roboty průmyslové, užívané ve výrobě, a roboty servisní, užívané při obslužných činnostech například domácí práce, zdravotnictví, údržba, průzkum, či armáda, viz obr. 5.

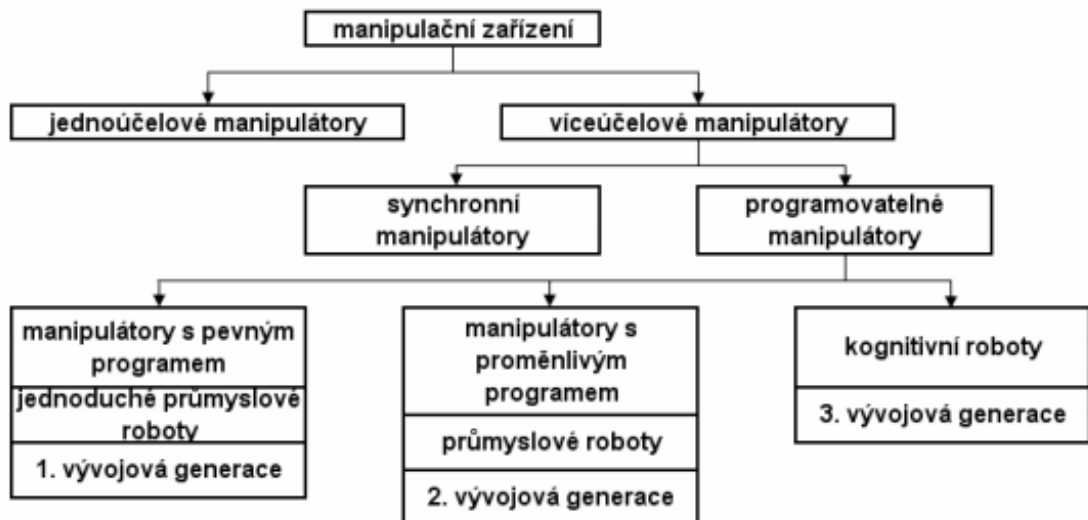
Nejvýznamnějším kritériem je možnost řízení a programování robotů. Základní rozdělení je znázorněno na obrázku 6.

Prvním skupinou jsou jednoúčelové manipulátory označované také jako podavače. Tyto zařízení se jen výjimečně vyskytují jako samostatné automatizační prvky. Obvykle jsou součástí obsluhovaného stroje, kde vykonávají jednu předdefinovanou funkci a jsou na něm konstrukčně a energeticky závislé [2].



Obr. 5 - Servisní průzkumný a pyrotechnický robot²

²SKAŘUPA, Jiří. *Průmyslové roboty a manipulátory*. Ostrava: Ediční středisko VŠB – TUO, 2007, ISBN 978-80-248-1522-0.



Obr. 6 – Rozdělení manipulačních zařízení³

Druhou skupinou jsou víceúčelové neboli univerzální manipulátory. Mají širší rozsah manipulačních schopností a s tím spojenou vyšší úroveň řízení. Univerzální manipulátory lze rozdělit ještě na manipulátory synchronní a programovatelné.

Synchronní manipulátory, označované také jako teleoperátory nebo exoskeletony, jsou řízeny průběžně pracovníkem. Tvoří společně se svým operátorem uzavřenou regulační smyčku. Manipulátor zesiluje silové a pohybové veličiny vyvolané člověkem. Umožňuje vzdálený přístup k místům, která jsou pro člověka nepřístupná nebo nebezpečná.

Programovatelné manipulátory jsou řízené vnitřním programovým ústrojím. Skupina je velice obsáhlá a dělí se do několika dalších podskupin. Manipulátory s pevným programem svůj řídicí program během chodu nemění. Manipulátory s proměnlivým programem mají schopnost přepínání a volby programu během procesu a tím reagují na změny. Tyto stroje vykazují značnou samostatnost a označujeme je jako průmyslové roboty. Posledním stupněm jsou roboty se schopností vnímat a samostatně se rozhodovat. Tato skupina je označována jako kognitivní roboty.

³Rozdělení manipulačních zařízení. In: *Vysoké učení technické v Brně* [online]. © 2016 VUT v Brně [cit. 18.8.2016]. Dostupný na: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=16454

3. Arduino

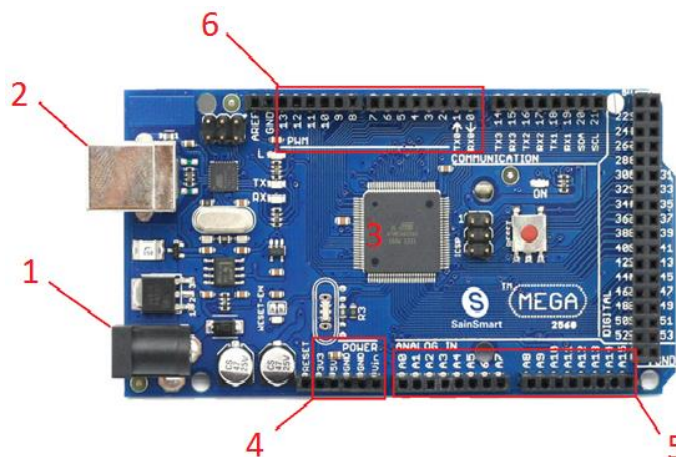
Hlavním cílem mé práce je vytvoření řídicího programu na ovládání polohy Teach-robotu. K tomuto úkolu je zapotřebí elektronické řídicí jednotky, která zprostředkuje komunikaci mezi robotem a operátorem. Nejvhodnější volbou jsou desky od výrobce Arduino [8]. Jedná se o otevřenou elektronickou platformu s uivatelsky jednoduchým hardware a software. Umožňuje propojit s dalšími hardware jako je například motor shield, který zde bude využit, případně provádět změny přímo na desce. Prostřednictvím platformy Arduino lze řídit další elektroniku, což v této práci bude potřeba k roztáčení motorů a načítání signálů ze snímačů.

3.1. Arduino Mega 2560

Pro potřeby Teach-robotu je využito platformy Arduino mega 2560 [9]. Na obrázku 7 je zobrazena deska a na ní vyznačené oblasti, na které budu dále odkazovat. Deska může být napájena v rozsahu 7-12V DC prostřednictvím adaptéru (1) nebo USB kabelu (2) propojeného s PC. Přes USB kabel se deska ovládá a jsou do ní nahrávány programy. Na středu desky je umístěn mikroprocesor Atmel ATmega2560(3) s frekvencí 16MHz, flash paměť 256kB, RAM paměť 8kB a EEPROM paměť 4kB.

Deska disponuje množstvím pinů, které umožňují propojení s další elektronikou a každý z nich má svůj účel a nelze je zaměňovat. Vyskytují se zde piny obstarávající zdroj napětí a uzemnění (4). Na této desce máme výstup pro 5V a 3.3V zdroj. Pod těmito piny se nachází 16 analogových vstupních pinů (5).

Deska obsahuje 54 digitálních vstupně-výstupních pinů kde pro 1(HIGH) můžeme na pinu naměřit napětí 5V a proud 40 mA. Mezi těmito piny zde bude důležité především 14 pulsně šířkových pinů PWM (6). U analogových i digitálních pinů je umístěno uzemnění GND.

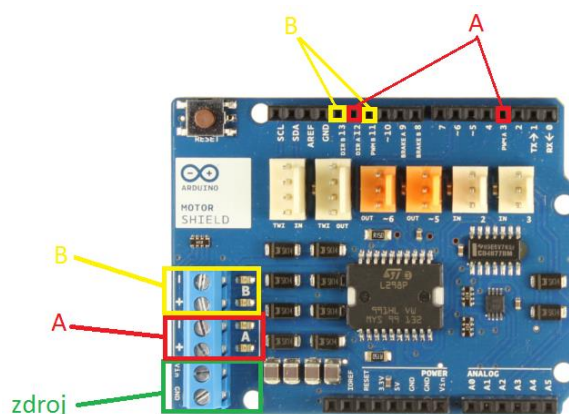


Obr. 7 - Arduino Mega2560⁴

⁴Arduino Mega2560. In: *Automatizace a robotizace Santy.cz* [online]. © Santy.cz [cit. 18.8.2016]. Dostupný z: <http://www.santy.cz/arduino-c2/arduino-mega2560-r3-i81/>

3.2. Motor Shield

I přes rozsáhle možnosti použití mají základní Arduino desky svá omezení. Pro tyto případy je možné propojit je s dalšími prvky, které tyto potřebné složitější úlohy dokáží vykonat. Těmto deskám se říká Shildy a mezi základní patří například Ethernet Shield, Wifi Shield nebo Motor Shield [7]. Pro práci s robotem bude potřeba Motor Shield a to rovnou v trojnásobném obsazení, jelikož každý Shield je schopen obsluhovat pouze dva motory současně. Tyto moduly lze zasunout do základní desky a piny na Shieldu přebírají vlastnosti odpovídajících pinů na desce. K tomu přidávají své vlastní vstupy a výstupy, které rozšiřují možnosti ovládání. V případě Motor Shieldu je to vstup pro externí napájení a dvojice výstupů jako zdroj proudu pro pohánění motorů. Pro řízení směru a rychlosti otáčení motorů jsou určeny dvě dvojice pinů, kde pin 3 (PWM) řídí rychlost pro motor na výstupu A, pin 12 (DIR) mění směr otáčení motoru A. Stejným způsobem fungují piny 11 (PWM) a 13 (DIR) pro motor na výstupu B.



Obr. 8 – Motor Shield⁵

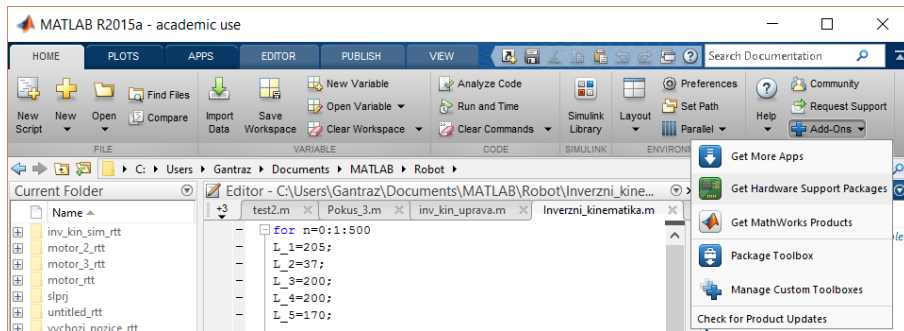
3.3. Simulink support package for arduino hardware

Platformy Arduino disponují vlastním programovacím jazykem, pro potřeby této úlohy je však zapotřebí zajistit programování a řízení z prostředí Matlab/Simulink. Předpokladem je nainstalované prostředí Matlab. Pro správnou komunikaci Arduina se Simulinkem je nezbytné doinstalovat patřičné knihovny, které přidají do Simulinku potřebné ovládací prvky.

3.3.1. Instalace knihovny Arduino

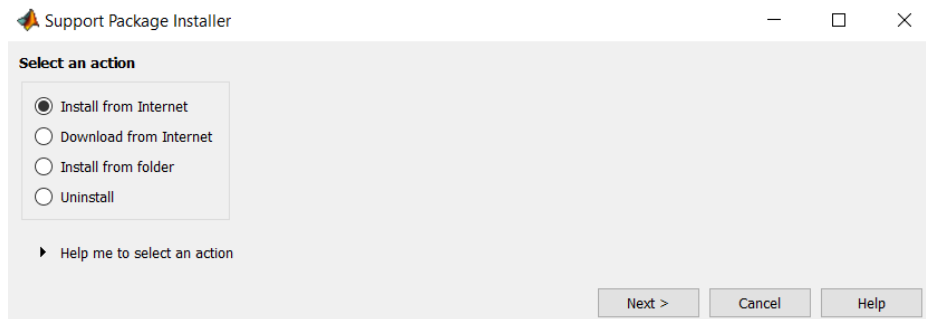
Nejdříve spusťte prostředí Matlab a otevřete položku *Add-Ons* v pravém horním rohu. Vyberte položku *Get Hardware Support Packages*, Obr. 9.

⁵ Motor Shield. In: *Arduino* [online]. ©2016 Arduino [cit. 18.8.2016]. Dostupný z: <https://www.arduino.cc/en/Main/ArduinoMotorShieldR3>



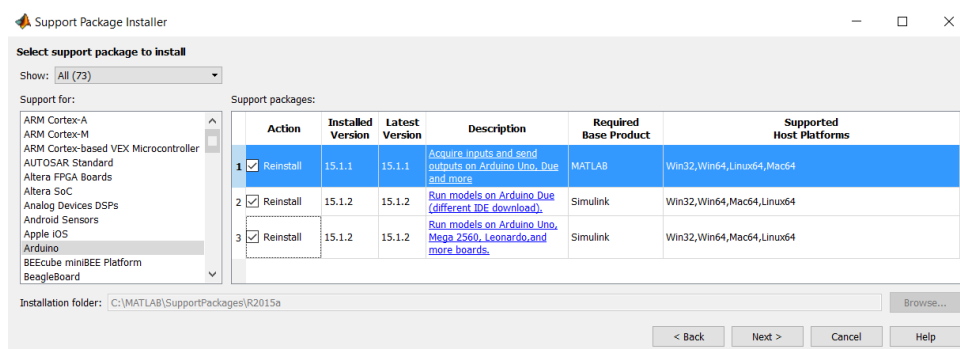
Obr. 9 – Instalace knihovny Arduino

Otevře se okno *Support Package Installer*, Obr. 10. Vyberte možnost *Install from Internet* a klikněte na tlačítko *Next*.



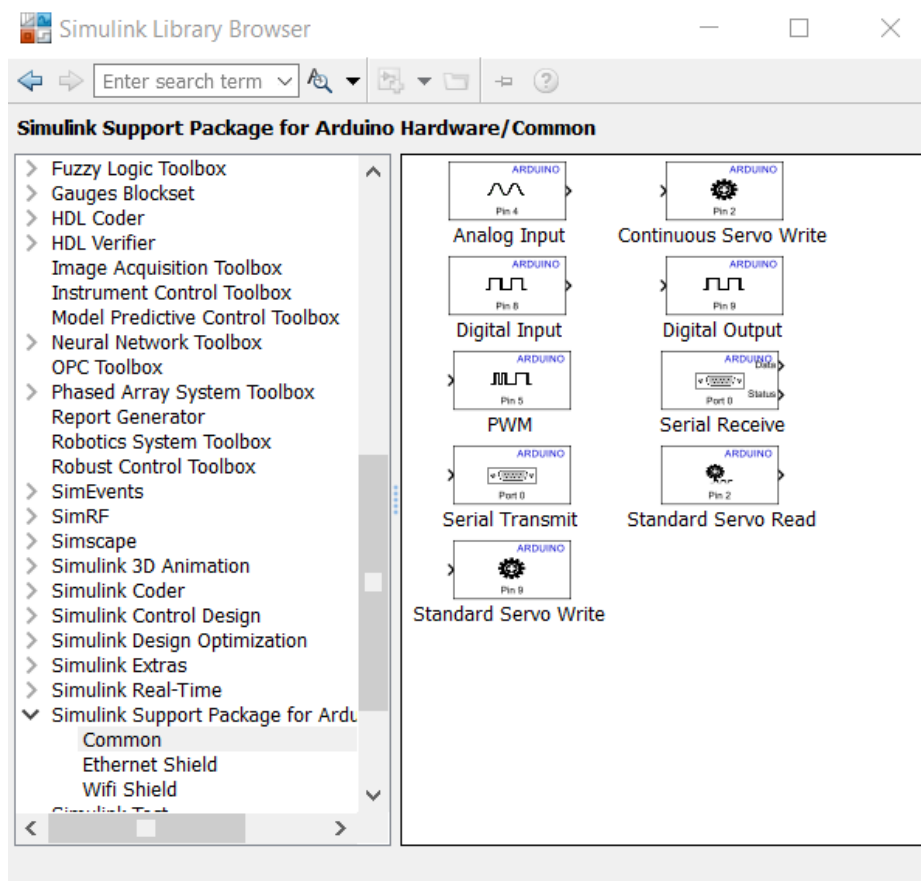
Obr. 10 - Support Package Installer

V následujícím okně vyhledejte podporu pro Arduino a označte všechny dostupné knihovny, Obr. 11. Pro dokončení instalace se musíte přihlásit k vašemu účtu MathWorks.



Obr. 11 – Výběr knihoven

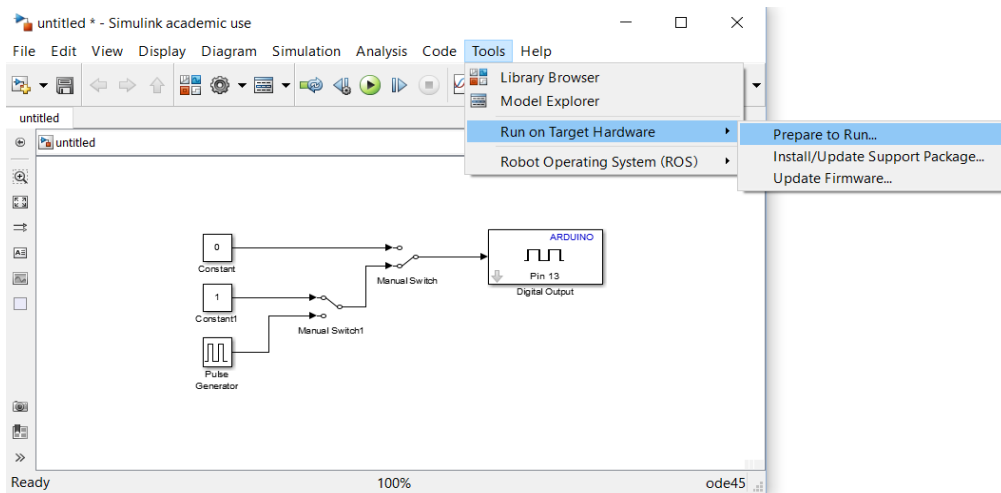
Po úspěšné instalaci se v seznamu knihoven pro Simulink objeví knihovna *Simulink Support Package for Arduino Hardware* a bude obsahovat několik skupin ovládacích bloků, z nichž je potřebná sada *Common*, obsahující bloky pro vstupy a výstupy digitálních pinů na Arduino, Obr. 12.



Obr. 12 – Knihovna Arduino

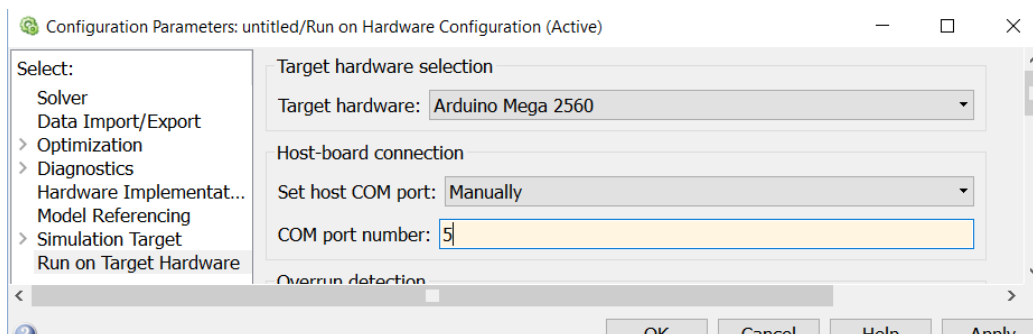
3.3.2. Nahrání programu do Arduino

Řídicí program musí být nahrán do paměti desky Arduino a až poté může být spuštěn. Pro ověření správné komunikace Simulinku s Arduinem jsem vytvořil jednoduchý program na ovládání LED diody umístěné na Arduino na pinu číslo 13. Aby se program nahrál na desku, je zapotřebí otevřít na záložku *Tools* → *Run on Target Hardware* → *Prepare to Run...*, Obr. 13.



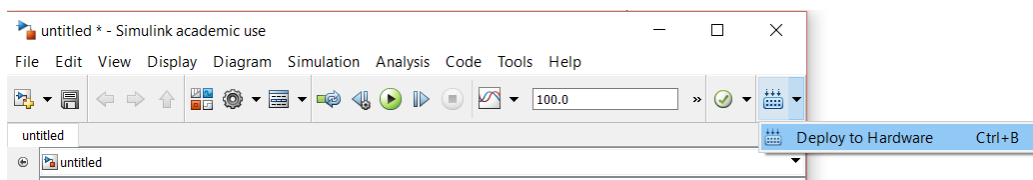
Obr. 13 – Ovládání diody

Po otevření okna *Configuration Parameters*, Obr. 14, je nutné zvolit cílový Hardware, v tomto případě je vybráno Arduino Mega 2560. Nastavení COM portu se provede manuálně a vybere se odpovídající port, na kterém je deska připojena. Tato informace je k dohledání ve správci zařízení.



Obr. 14 – Configuration Parameters

Nakonec zbývá už jen nahrání programu do Arduina. Nastaví se simulační čas a zvolí externí mód. Tlačítkem *Deploy to Hardware* v pravém horním rohu se nahraje program na desku a tlačítkem *Run* lze program spustit, Obr. 15.



Obr. 15 – Spuštění programu

4. Teach-robot

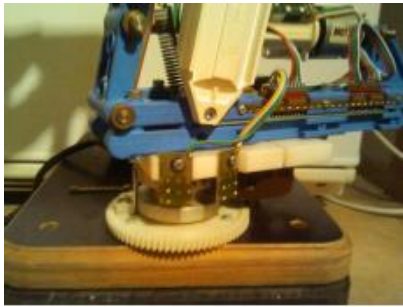
Předmětem této práce je Teach-robot. V následující kapitole se seznámíme s jeho konstrukčním provedením. Původně se tento typ vyskytoval na ústavu ve dvou exemplářích, bohužel oba kusy utrpěli poškození, které je vyřadilo z provozu. Poškození naštěstí nebylo příliš rozsáhlé a netýkalo se shodných konstrukčních prvků těchto robotů. To umožnilo s využitím dílů obou nefunkčních kusů sestavit jeden provozuschopný stroj. Tímto úkolem se zabývala práce pana Petra Šťastného [3], na kterou budu v této kapitole odkazovat. Vzhledem k absenci originální technické dokumentaci Teach-robotu, byla tato práce jediný dostupný materiál popisující technické provedení robotu.

4.1. Konstrukční provedení

Teach-robot spadá do skupiny servisních manipulátorů. Má sériovou kinematickou strukturu s pěti stupni volnosti. Pohon je zprostředkován šesti elektromotory. Otáčení celého robotu kolem svislé osy zajišťuje jeden radiální pohon umístěný v podstavě celé konstrukce, Obr. 16.

Další radiální pohon se nachází na konci robotického ramene, k zajištění rotace zápěstní části. Natočení hlavních pohybových kloubů a náklon jednotlivých segmentů robotu obstarávají tři lineární pohony. Jejich uchycení je znázorněno na obrázku 17. Tření v převodovém mechanismu dostačuje k udržení vlastní váhy a zajistí tedy, že po odpojení napájení motorů se robot nezhroutí, ale zůstane přesně v pozici, kde byl před přerušením dodávky proudu. To umožňuje vypnutí pohonů po dosažení požadované pozice bez nutnosti použití přídatného brzdného mechanismu nebo vyvolání síly v motorech potřebné k překonání vnějších sil působících na konstrukci. Poslední motor slouží k otevírání a zavírání úchopového mechanismu pomocí lineárního jezdce s drátěným spojem na kleštinu.

Ovládání je zde řešeno pomocí řídicího panelu napojeného na sběrnici signálů ze všech částí robotu vybaveného párem spínačů pro každý motor na řízení otáčení v jednom nebo druhém směru, Obr. 18. Jedná se tedy o řízení zcela závislé na operátorovi bez možnosti automatizace. Lze jej tedy zařadit do skupiny synchronních robotů. Po vytvoření ovládacího programu bude tento panel nahrazen platformou Arduino, která po propojení s PC umožní řízení všech motorů současně zadáváním buď požadovaných úhlů natočení, nebo cílových souřadnic pracovního bodu. Tímto způsobem umožníme automatizaci pohybu Teach-robotu.



Obr. 16 – Radiální motor u podstavy robotu⁶



Obr. 17– Uchycení lineárních pohonů⁶



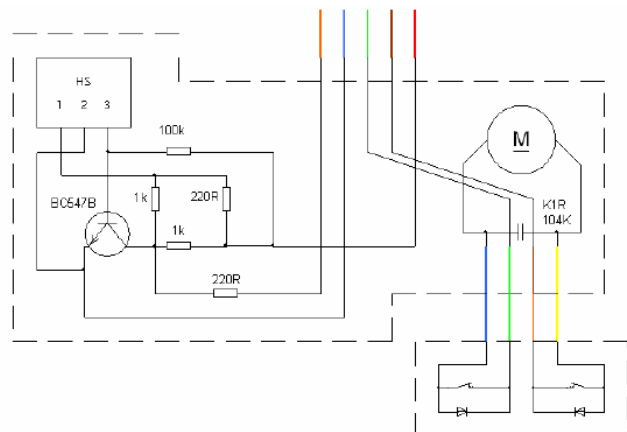
Obr. 18– Teach-robot ovládací panel⁶

4.2. Provedení elektronických obvodů

K zajištění bezproblémového chodu motorů a jejich správného připojení na řídicí desku je nutná znalost provedení elektrických obvodů. Na výstupu z každého motoru je vedeno pět drátů s barevným rozlišením. Na obrázku 19 je zakresleno schéma elektronického obvodu pro jeden motor.

4.2.1. Zapojení motorů

Napájení motoru je přiváděno zeleným a hnědým drátem, které jsou vedeny přes koncové spínače přerušující obvod při jejich sepnutí, dojde-li rameno do své krajní polohy. Toto opatření zabraňuje poškození pohonů a konstrukce robota vycházejícího z případného špatného řízení. Motory se jednoduše vypnou a nedojde tedy k jejich zbytečnému přehřívání při snaze natočit rameno mimo konstrukční možnosti robota. Motory jsou určeny k napájení zdrojem napětí 19 V. Při pokusech jsem si vystačil i s nižším zdrojem 12 V. Ten byl přiveden k motorům přes Platformu Arduino pomocí Motor-Shieldu, jelikož základní deska je stavěna pouze pro maximální napětí 5 V.



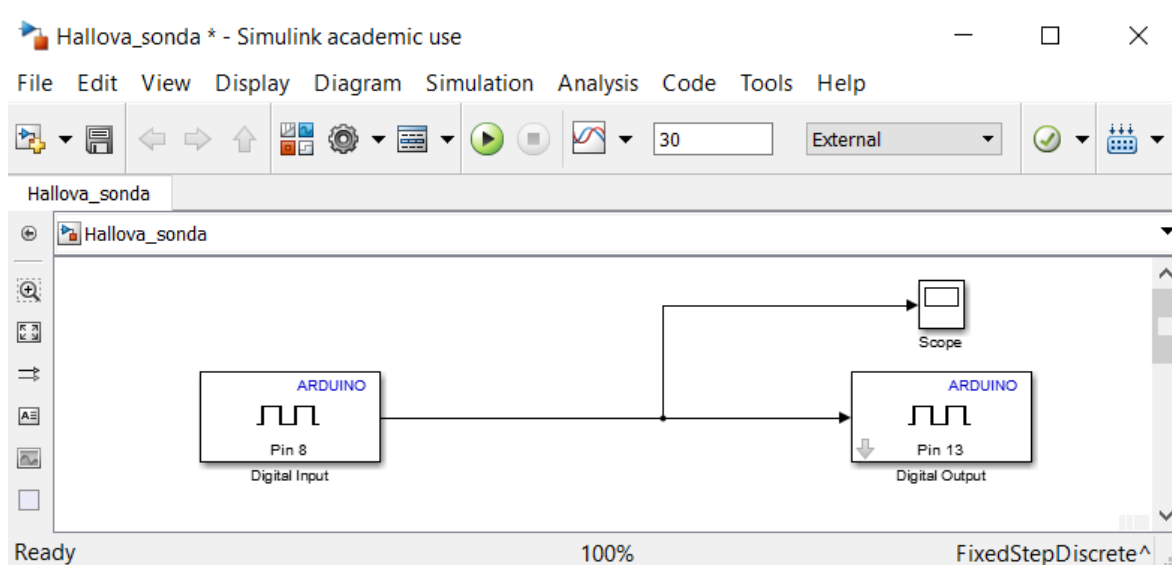
Obr. 19 – Elektronický obvod⁶

⁶ŠŤASTNÝ, Petr. *Příprava výukového robota*. Praha: ČVUT 2013. Bakalářská práce, ČVUT, Fakulta strojní, Ústav mechaniky, biomechaniky a mechatroniky.

4.2.2. Hallova sonda

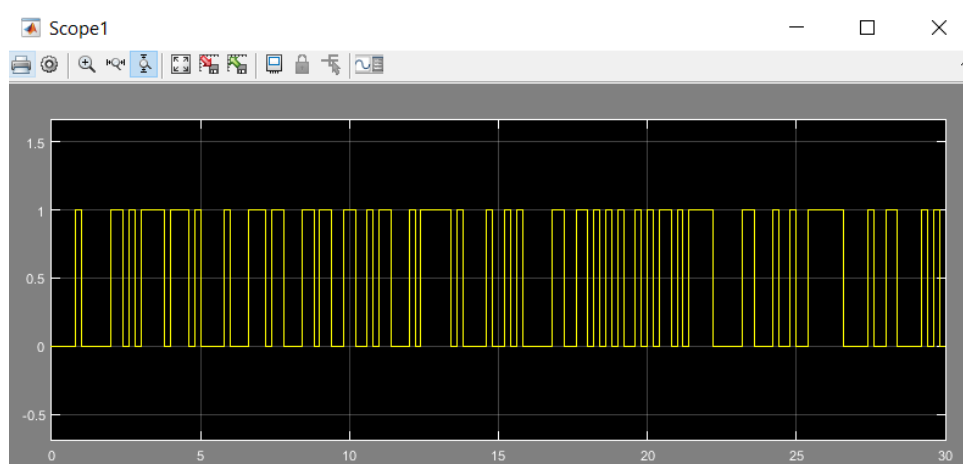
Zbývající tři dráty zajišťují zapojení Hallovy sondy. Červený drát dodává potřebné napětí a bude vyveden k základní desce Arduina jejichž 5V zdrojový pin dostačuje nárokům na napájení sondy. Uzemnění je svedeno po modrém drátě a taktéž povede do základní desky tentokrát na pin GND. Poslední bílý drát (na schématu znázorněn oranžovou barvou) slouží k vyvedení signálu, jenž poskytuje senzor. Tento údaj umožní určit aktuální natočení motorů a skutečnou polohu ramen.

Senzor Hallova sonda zaznamenává změny magnetického pole. Ty se projeví změnou napětí. Hallova sonda na motorech Teach-robot je vybavena čtyřmi magnety, které na jednu otočku motoru vytvoří čtyři pulzy. Tyto pulzy jsou svedeny na bílý drát a dají se snadno zaznamenat. Pro čtení těchto signálu jsem vytvořil jednoduchý program v Simulinku, Obr. 20. Hallovu sondu jsem napájel z Arduina a signál ze senzoru připojil na pin 8. V Simulinku jsem vzal z knihovny Arduina bloček pro vstup a přiřadil mu čtení pinu 8. Signál jsem svedl blokem pro výstup také na pin 13, který je svázaný s diodou a při každém zaznamenaném pulsu ji rozsvítil. Zaznamenané pulzy jsem nechal vykreslovat do grafu, Obr. 21. Arduino jsem propojil s PC a spustil program. Ručním otáčením motoru jsem vybudil pulzy na sondě a zaznamenal měření.



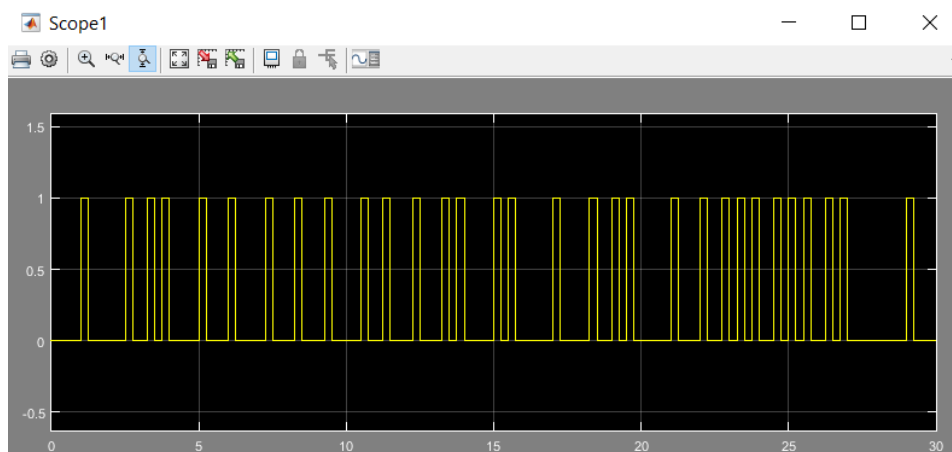
Obr. 20– Načítání Hallovy sondy

Když sonda zaznamená pulz, vypíše na výstupu hodnotu 1. Pro načítání množství pulzů toto není zcela vhodné. Zastaví-li se sonda na hodnotě 1, bude program při každém následujícím cyklu přičítat hodnotu tohoto signálu, dokud se motor neotočí a výstupní signál nebude na hodnotě 0.



Obr. 21 – Načítání Hallové sondy, osa x – čas v sekundách, osa y - signál sondy

Aby program zaznamenal každý pulz pouze jednou, je třeba ošetřit výstupní signál pomocí funkce *Detect increase*, která vrací hodnotu 1 pouze, zaznamená-li nárůst vstupního signálu a to po dobu jednoho cyklu programu. V ostatních případech vrací hodnotu 0, Obr. 22.

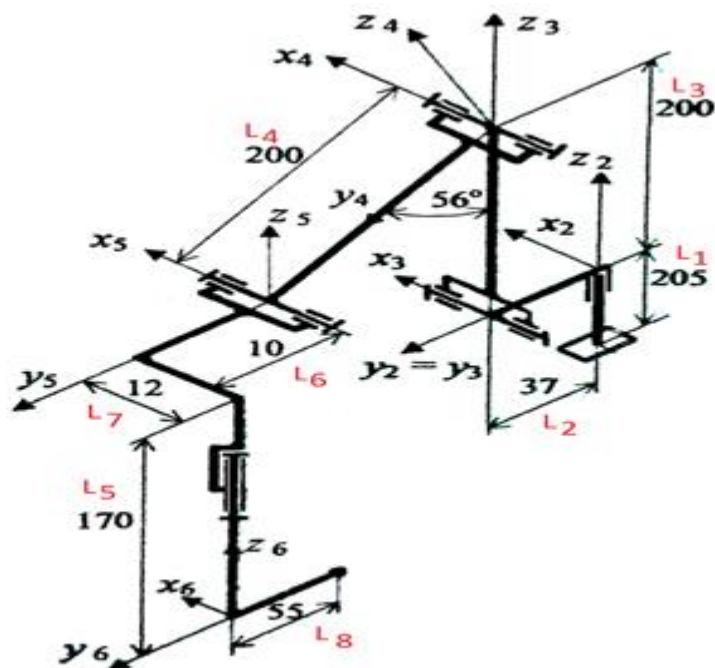


Obr. 22 – Ošetřený výstupní signál sondy, osa x – čas v sekundách, osa y - signál sondy

S takto ošetřeným signálem bude později možné načítat jednotlivé pulzy a určit natočení motoru.

5. Kinematický model Teach-robotu

Pro práci s robotem je nutné definovat jeho kinematický model. Ten poslouží k určení řízených veličin tak, aby se dosáhlo požadovaného koncového stavu. Zde je na výběr ze dvou variant kinematických modelů, které se ve výsledku liší možnostmi řízení. Dopředná a inverzní kinematická úloha. V této kapitole rozeberu obě varianty, přičemž druhá zmíněná je z hlediska řízení významnější. Schéma a rozměry konstrukce Teach-robotu jsou zobrazeny na obrázku 23.



Obr. 23 – Schéma Teach-robotu⁷

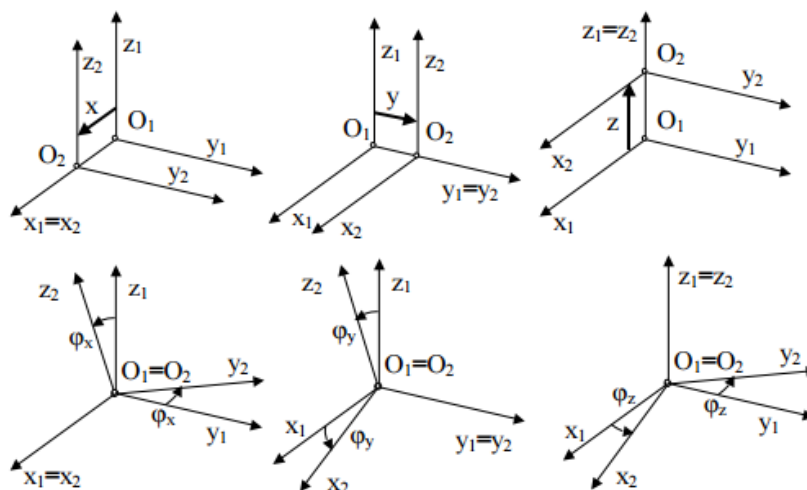
5.1. Dopředná kinematika

První možností je úloha dopředné kinematiky. Tato úloha vychází ze znalosti natočení všech pohonů a jejím cílem je určení polohy koncového bodu vůči počátku. Pro samotné řízení však tento model není zcela vhodný.

Nejvhodnější metodou k vyřešení dopředné kinematické úlohy Teach-robotu je maticová metoda. Tato metoda využívá přechodu mezi souřadnicovými systémy všech segmentů od výchozího bodu postupně až k bodu, jehož souřadnice jsou požadovány. Pro popis přechodu z jednoho systému na druhý slouží transformační matice a tyto přechody je možné řetězit za sebe, dokud se nedosáhne koncového bodu. Pro každé těleso je zapotřebí

⁷převzato z učebních materiálů ústavu mechaniky, biomechaniky a mechatroniky

zavést vlastní kartézský souřadnicový systém. Poté lze přistoupit k popisu transformace mezi souřadnicovými systémy. Lze definovat šest základních pohybů souřadnicových systémů, Obr. 24 [4].



Obr. 24 – Základní pohyby⁸

Transformační matice popisující těchto šest základních pohybů jsou čtvercové matice a pro těleso v prostoru mají rozměr 4x4. Lze je rozdělit na posuvy podél osy a rotace kolem osy, které jsou popsány maticí směrových kosinů. Transformační matice se značí písmenem **T** pro posuv, případně **T_φ** pro rotaci a indexem označujícím vůči jaké ose je pohyb prováděn.

$$T_x(x) = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$T_y(y) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$T_z(z) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

⁸VALÁŠEK, Michael. *Kinematika robotických systémů* [online]. ČVUT Praha 2011 [cit. 21.8.2016]. Dostupný z: http://www.crr.vutbr.cz/system/files/brozura_06_1102.pdf

$$T_{\varphi_x}(\varphi_x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \varphi_x & -\sin \varphi_x & 0 \\ 0 & \sin \varphi_x & \cos \varphi_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$T_{\varphi_y}(\varphi_y) = \begin{bmatrix} \cos \varphi_y & 0 & \sin \varphi_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \varphi_y & 0 & \cos \varphi_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

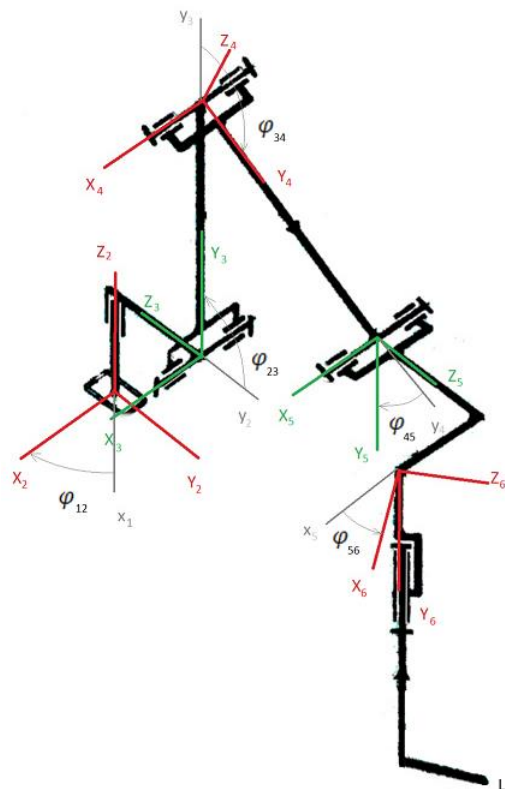
$$T_{\varphi_z}(\varphi_z) = \begin{bmatrix} \cos \varphi_z & -\sin \varphi_z & 0 & 0 \\ \sin \varphi_z & \cos \varphi_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

Nyní je vše připravené k popisu kinematického modelu Teach-robotu. Je nezbytné dodržet posloupnost jednotlivých kroků. Na obrázku 25 je vykresleno schéma robota a zvolené souřadnicové systémy jednotlivých segmentů.

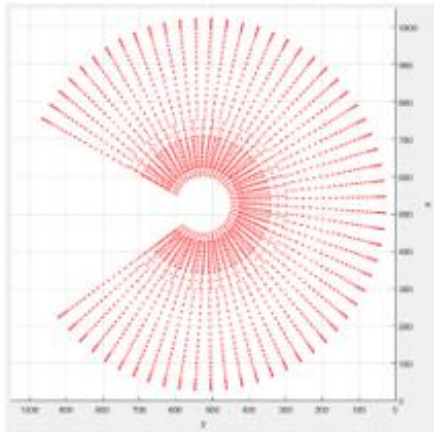
Poloha koncového bodu L:

$$r_{1L} = T_{\varphi_{z12}}(\varphi_{12}) \cdot T_{z23}(L_1) \cdot T_{y23}(L_2) \cdot T_{\varphi_{x23}}(\varphi_{23}) \cdot T_{y34}(L_3) \cdot T_{\varphi_{x34}}(\varphi_{34}) \cdot T_{y45}(L_4) \cdot T_{\varphi_{x45}}(\varphi_{45}) \cdot T_{z56}(L_6) \cdot T_{x56}(L_7) \cdot T_{\varphi_{y56}}(\varphi_{56}) \cdot \begin{pmatrix} 0 \\ L_5 \\ L_8 \\ 1 \end{pmatrix} \quad (7)$$

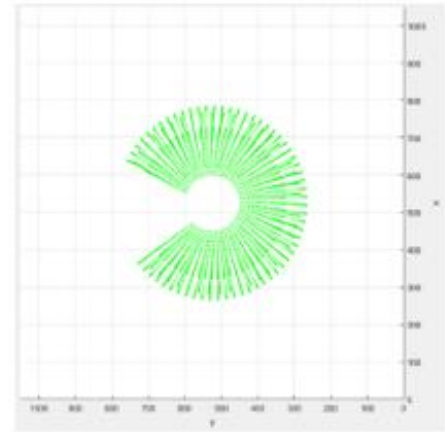
Po sestavení rovnice na výpočet souřadnice koncového bodu je vytvořen kinematický model, se kterým je možné dále pracovat. S využitím této metody jsem napsal program v Matlabu na výpočet všech významných bodů mechanického modelu a vytvořil digitální model Teach-robotu [5]. Na Teach-robotu jsem provedl měření maximálních a minimálních rozsahů natočení kloubů. S těmito poznatky bylo možné za pomoci programu na dopřednou kinematiku vykreslit pracovní prostor Teach-robotu. Na obrázcích 26 až 31 je zobrazený digitální model robota a pracovního prostoru.



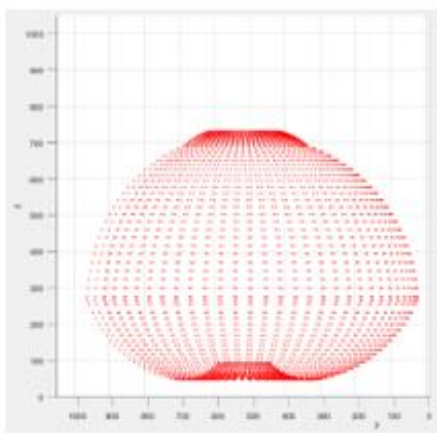
Obr. 25 – Transformace souřadnicových systémů



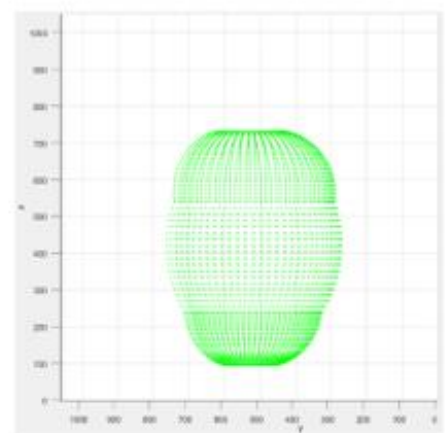
Obr. 26 – Maximální dosah XY



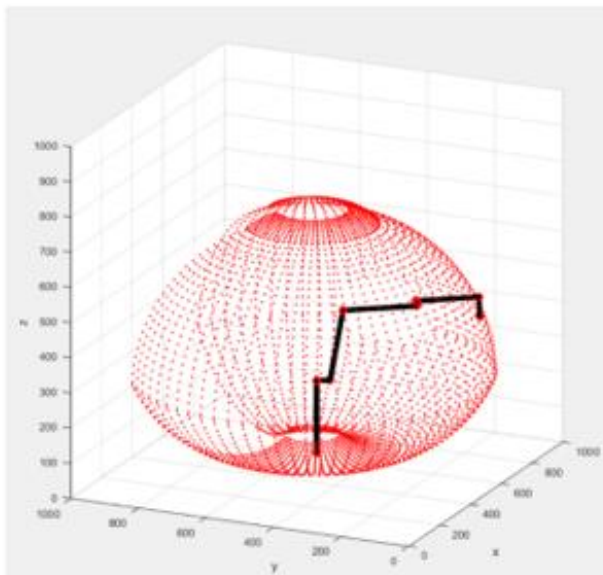
Obr. 27 – Minimální dosah XY



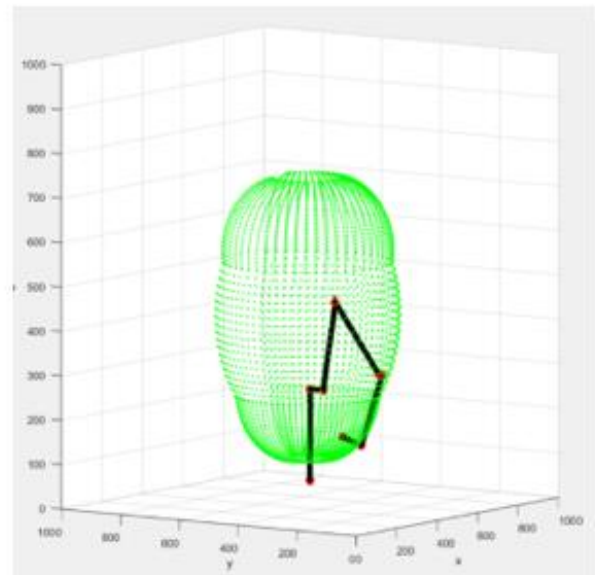
Obr. 28 – Maximální dosah YZ



Obr. 29 – Minimální dosah YZ



Obr. 30 - Maximální rozsah 3D



Obr. 31 - Minimální rozsah 3D

5.2. Inverzní kinematika

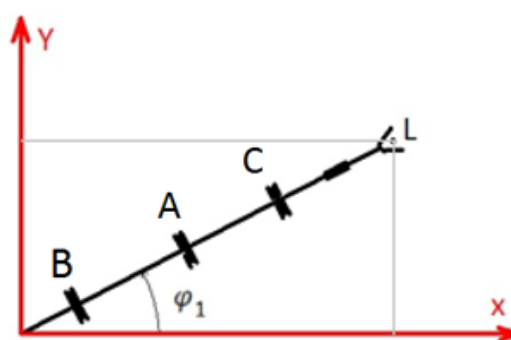
Pro samotné řízení je mnohem významnější model inverzní kinematiky. Tento model vychází z předpokladu, že je známá pozice posledního bodu mechanismu, v tomto případě poloha úchopového ústrojí na konci ramene, a je zapotřebí určit polohu všech zbývajících částí robotu tak, aby bylo této pozice skutečně dosaženo. Tato úloha více odpovídá reálným potřebám pro řízení, kdy je známá poloha robotu a jsou určeny cílové souřadnice, na které je třeba se přesunout, například při přenášení nákladu z místa **A** na místo **B**, případně při zadání množiny bodů na projetí určité trajektorie při svařování a řady dalších pracovních úkonů.

U složitějších mechanických struktur, které se nedají vyřešit analytickou cestou, se využívá numerické řešení. Tento způsob řešení je závislý na výkonu výpočetního zařízení. Provádí výpočty stále dokola a s každým krokem zpřesňuje výsledek, dokud se nedostane pod hodnotu maximální povolené odchylky. S vyššími požadavky na přesnost se prodlužuje potřebný čas na výpočty a obráceně s větším nárokem na rychlost lze očekávat výraznější odchylky od skutečné požadované pozice. Tato metoda není zcela vhodná pro řízení v reálném čase.

V případě Teach-robotu se nejedná o příliš složitou konstrukci a je tedy možné nalézt řešení pomocí analytické metody. Celý mechanismus lze promítnout do jedné roviny a dále vypočítat natočení této roviny v prostoru. Nejprve budu řešit natočení robotu vůči rámu. Zavedu souřadnicový systém a natočím rameno do obecné polohy. Z obrázku 32 je patrné, že výsledný úhel lze vypočítat přímo ze zadaných souřadnic koncového bodu.

$$\operatorname{tg} \varphi_1 = \frac{y_L}{x_L} \quad (8)$$

$$\varphi_1 = \operatorname{arctg} \left(\frac{y_L}{x_L} \right) \quad (9)$$



Obr. 32 – Natočení roviny robotu

Nyní je známá pozice koncového bodu **L** a natočení roviny ramena. Pro řízení je vhodné mít možnost zvolit, v jakém úhlu vůči podstavě bude robot uchopovat předměty. Operátor tedy bude volit souřadnice koncového bodu a sklon čelistí. Z těchto dat tedy lze dopočítat i polohu posledního (zápěstního) kloubu. Tento bod označím písmenem **C**, Obr. 33.

V rovině ramene zavedu souřadnici **R**, která popisuje promítnutou délku segmentů do roviny **XY**. Polohu bodu **C** určíme z následujících rovnic.

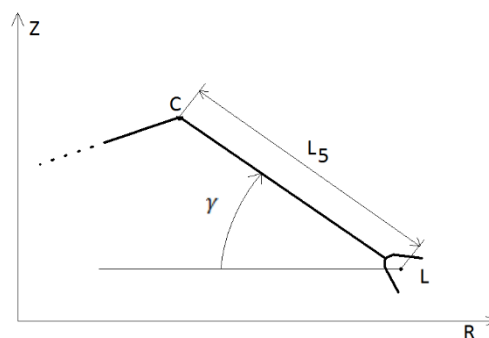
$$R_L = \sqrt{x_L^2 + y_L^2} \quad (10)$$

$$R_C = R_L - L_5 * \cos \gamma \quad (11)$$

$$z_C = z_L + L_5 * \sin \gamma \quad (12)$$

$$x_C = R_C * \sin \varphi_1 \quad (13)$$

$$y_C = R_C * \cos \varphi_1 \quad (14)$$



Obr. 33 – Sklon čelistí

Pozice prvního kloubu je závislá pouze na natočení celého ramena. Výška z_B odpovídá výšce L_1 a vzdálenost R_B je rovna délce L_2 , viz Obr. 23.

$$z_B = L_1 \quad (15)$$

$$x_B = L_2 * \sin \varphi_1 \quad (16)$$

$$y_B = L_2 * \cos \varphi_1 \quad (17)$$

Poslední neznámá pozice je kloub **A**. Aby bylo možné získat reálné výsledky, je zapotřebí ošetřit rozsah pohybů a dosah ramena. Vzdálenost mezi bodem **B** a **C** nesmí být větší, případně menší než jakou jsou schopny propojit druhý a třetí segment. V případě že pozice koncového bodu přesahuje za hranice dosahu, je nutné přepočítat vzdálenost bodu **C** od hlavní osy tak, aby jej vzhledem ke konstrukci robota bylo možné dosáhnout. Následně se převrácením rovnice (11) a (12) dopočítá poloha koncového bodu.

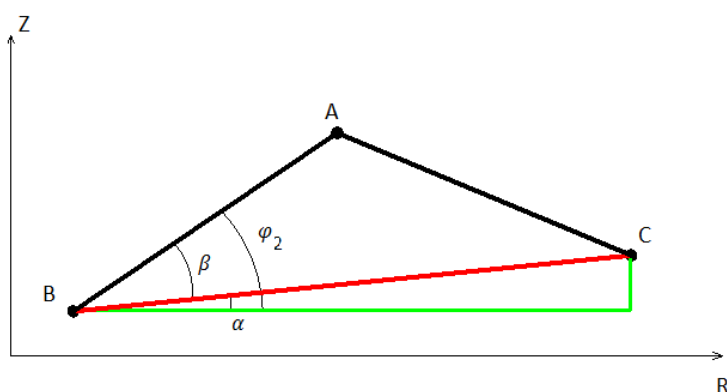
Limitní údaje jsou délka druhého a třetího segmentu společně s maximálním a minimálním úhlem, který mezi sebou svírají. S využitím kosinové věty lze určit rozsah možné vzdálenosti bodů **B** a **C**.

$$L_{BCmax} = \sqrt{L_3^2 + L_4^2 - L_3 * L_4 * \cos \varphi_{3max}} \quad (18)$$

$$L_{BCmin} = \sqrt{L_3^2 + L_4^2 - L_3 * L_4 * \cos \varphi_{3min}} \quad (19)$$

$$L_{BC} = \sqrt{(R_C - R_B)^2 + (z_C - z_B)^2} \quad (20)$$

Překročí-li hodnota L_{BC} vymezený rozsah, nahradí se tato hodnota limitní hodnotou a úpravou rovnice (20) za předpokladu ponechání stejné výšky dojde k přepočtu souřadnice R_C . Tím je zajištěna možnost reálného výsledku pro výpočet bodu A. Při sestavování rovnic budu vycházet z obrázku 34.



Obr. 34 - Poloha kloubu A

$$\alpha = \arctg \frac{z_C - z_B}{R_C - R_B} \quad (21)$$

$$\beta = \arccos \frac{(L_4^2 - L_3^2 - L_{BC}^2)}{-2 * L_3 * L_{BC}} \quad (22)$$

$$\varphi_2 = \alpha + \beta \quad (23)$$

$$R_A = R_B + L_3 * \cos \varphi_2 \quad (24)$$

$$z_A = L_1 + L_3 * \sin \varphi_2 \quad (25)$$

$$x_A = R_A * \cos \varphi_1 \quad (26)$$

$$y_A = R_A * \sin \varphi_1 \quad (27)$$

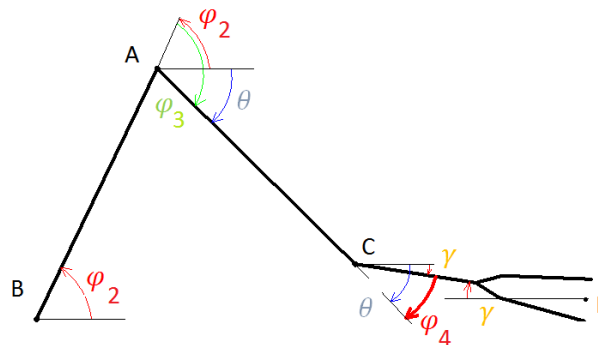
V tuto chvíli je definovaná poloha všech kloubů robotu. Pro řízení ovšem znalost souřadnic kloubů nemá žádný užitek. Pohyb je vyvolán otáčením motorů, které mění úhel sklonu mezi jednotlivými segmenty. Natočení motorů jsem schopen odměřovat jak bylo

vysvětleno v kapitole 4.2.2. Zbývá tedy ještě určit vztah mezi úhly, které mezi sebou svírají jednotlivé segmenty a výslednou polohou robotu.

Dosud je definovaný vztah pro natočení celého ramene kolem hlavní osy (9) a naklopení prvního segmentu (21)-(23). Pro natočení třetího členu vůči druhému budu vycházet z obrázku 34. Využiji znalosti vzdálenosti mezi body **B** a **C** a pomocí kosinové věty určím potřebný úhel. Poslední úhel lze určit podle obrázku 35.

$$\varphi_3 = 180 - \arccos \frac{(L_{BC}^2 - L_3^2 - L_4^2)}{-2 * L_3 * L_4} \quad (28)$$

$$\varphi_4 = \varphi_2 - \varphi_3 - \gamma \quad (29)$$



Obr. 35 – Úhel úchopového mechanismu

Tímto je vytvořený model pro popis inverzní kinematické úlohy. Tento postup jsem využil pro napsání programu v Matlabu a vytvořil tak obdobný virtuální model jako u dopředné úlohy, Obr. 30. V této fázi dovedu načítat polohu motorů a pro zadané souřadnice koncového bodu a sklon úchopového mechanismu vypočítat potřebné natočení motorů pro dosažení požadované polohy.

6. Řídicí program

Řízení bude zprostředkováno platformou Arduino, *kap. 3*, s využitím údajů o poloze řešených v předchozím oddíle, *kap. 5.2*. Pohon celého Teach-robotu tvoří šest stejných motorů, jejichž polohu jsem schopen určit z dat poskytnutých Hallovo sondou, *kap. 4.2.2*. V následující kapitole se budu zabývat roztáčením motorů, řízením jejich rychlosti a směru otáčení.

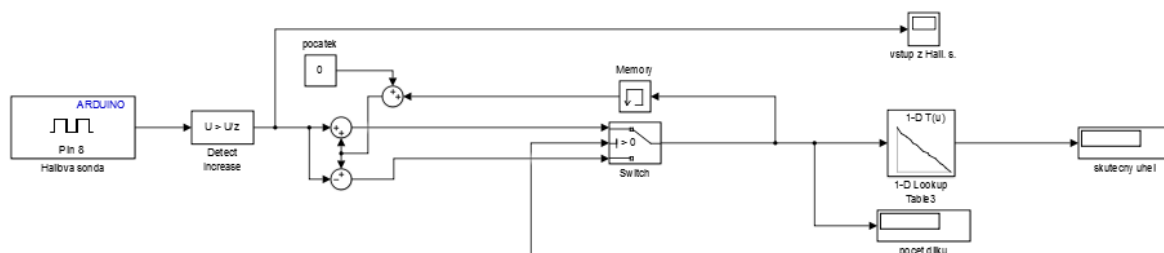
6.1. Ovládání motoru

Propojení motoru s řídicí deskou bylo popsáno v předchozích kapitolách *3.2 a 4.2.1*. V této části vytvořím pomocí Simulinku program na polohování motoru dle požadovaného úhlu natočení [6]. Program rozdělím do třech úseků. Určení polohy, nastavení směru otáčení a nastavení rychlosti otáčení. Tyto úseky budou na konci provázány a společně tvoří řízení jednoho motoru, kde vstupem bude žádaná poloha motoru a výstupem skutečné natočení motoru. Cílem je, aby se vstupní a výstupní hodnoty rovnaly.

6.1.1. Skutečná poloha motoru

Na obrázku 36 je usek programu, který zaznamenává a vyhodnocuje údaje z Hallovy sondy. Na vstupu je čtení signálů, které sonda posílá na pin 8. Tento signál je v následující bloku ošetřen jak bylo vysvětleno v kapitole 4.2.2. Výsledný signál je pro kontrolu vykreslen do grafu. Blok memory postupně nasčítává veškeré zaznamenané pulzy. Počáteční hodnota je nastavena na **0**. Senzor není schopen rozeznat kterým směrem se motor otáčí a tedy zda má zaznamenané pulzy přičítat či odčítat. Program vždy provede obě operace, do paměti však pošle vždy jen tu, která odpovídá aktuálnímu směru otáčení motoru. Program je třeba naladit pro každý motor tak, aby z výchozího bodu, kdy bude rameno ve své krajní poloze, ať už to je maximální nebo minimální úhel náklonu, začal s otáčením v kladném směru vzhledem k zaznamenaným pulzům a tedy aby počet nasčítaných pulzů byl v kladných hodnotách. Tím je vyřešeno zaznamenání polohy motoru. Model pro inverzní kinematiku však počítá nastavení úhlu mezi jednotlivými segmenty robotu. Zde je však zaznamenaný pouze počet pulzů. Je tedy nutné převést tyto pulzy na odpovídající hodnotu úhlu sklonu segmentů. Z důvodů konstrukčního provedení, tedy ze způsobů uchycení motorů na rameno robotu není převod pulzů na úhel lineární. Tento problém pomůže vyřešit blok *Lookup Table*. Tento blok provede převod pulzů na stupně podle funkce, kterou interpoluje z naměřených údajů. Odměřil jsem úhly po celém rozsahu pohybu s krokem 5 až 10 stupňů a k nim přiřadil odpovídající počet zaznamenaných pulzů. Naměřená data jsem zaznamenal do příslušného

bloku v programu. Nyní mám vytvořený úsek pro určení skutečného náklonu segmentů.

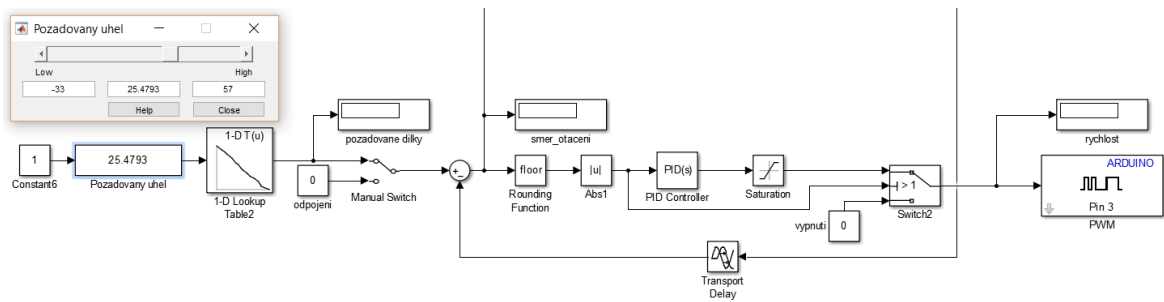


Obr. 36 – Poloha motoru.

6.1.2. Nastavení rychlosti motoru

Rychlost otáčení motoru závisí na velikosti napětí, které je do motoru přivedeno. To odpovídá hodnotě signálu, který je poslán na pin 3 případně na pin 12. Hodnota tohoto signálu je v rozmezí 0 až 255 přičemž pro hodnotu 0 je motor odpojen a pro hodnotu 255 je motor napájen maximálním možným napětí přiváděním ze zdroje. Úsek programu pro nastavení rychlosti je na obrázku 37.

Na vstup je přiveden požadovaný úhel. Program pracuje s počtem pulzů a je tedy nutné převést daný úhel na odpovídající počet pulzů. Tato hodnota je následně porovnána se skutečnou hodnotou naměřenou na motoru a vypočítá se rozdíl. Výsledkem je kladná nebo záporná hodnota. Ta rozhoduje o směru otáčení, více v kapitole 6.1.3. V následujících dvou blocích programu se rozdíl zaokrouhlí na celá čísla a převede na kladnou hodnotu. Následuje blok PID regulátoru. V případě malého rozdílu skutečné a požadované polohy může nastat, že signál přivedený na pin 3 neposkytne motoru dostatečné napětí a ten nebude mít dostatečnou sílu se roztočit. Proto je zapotřebí regulátoru s nenulovou integrační složkou, který bude v takovém případě navyšovat hodnotu výstupního signálu dokud nebude dostatečně silný na roztočení motoru. Pin 3 přijímá signál v rozsahu 0 až 255. Rozdíl požadované a skutečné polohy však může vyvolat výrazně vyšší hodnotu, proto je třeba před vstupem na pin 3 zredukovat velikost signálu. K tomuto účelu slouží blok *Saturation*. Poslední blok před zápisem signálu na pin slouží k odpojení motoru. Je-li rozdíl mezi skutečnou a žádanou polohou pod hranicí maximální dovolené odchylky, dojde k vypnutí motoru. Tím se zabrání nestabilitě motoru v těsné blízkosti žádané polohy. V tuto chvíli jsem schopen řídit otáčení motoru.



Obr. 37 – Spouštění motoru

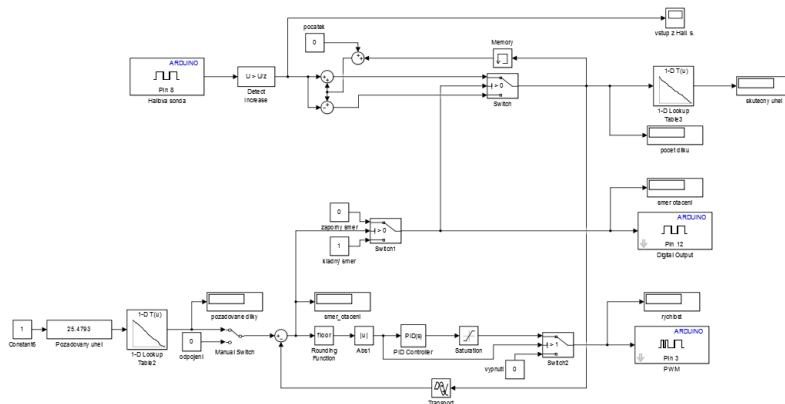
6.1.3. Nastavení směru otáčení

Z rozdílu žádané a skutečné hodnoty získám potřebné informace, abych mohl rozhodnout, který směr otáčení mě přiblíží k požadovanému cíli. Je-li výsledek záporný, skutečný úhel je větší než úhel požadovaný a je tedy potřeba záporný smysl otáčení. Při záporném rozdílu je třeba kladného smyslu otáčení. Úsek programu pro nastavení směru je na obrázku 38. Piny 12 a 13 rozhodují o směru otáčení motorů. Pracuje pouze s hodnotami **1** pro jeden směr a **0** pro druhý směr. Blok *Switch* vyhodnotí požadovaný směr a vyšle příslušný signál.



Obr. 38 – Směr otáčení

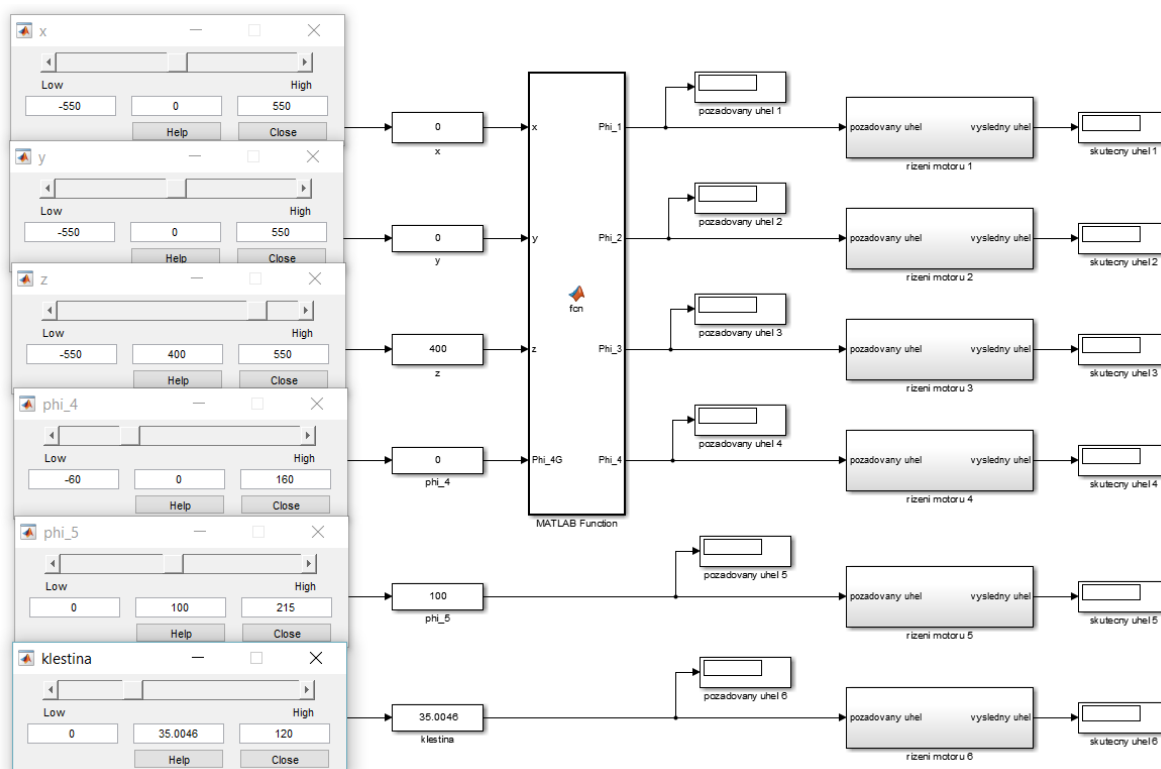
Složení všech tří úseků vznikne program na polohování jednoho motoru, Obr 39. Tento program jsem využil na řízení pokusného motoru.



Obr. 39 – Řídicí program motoru

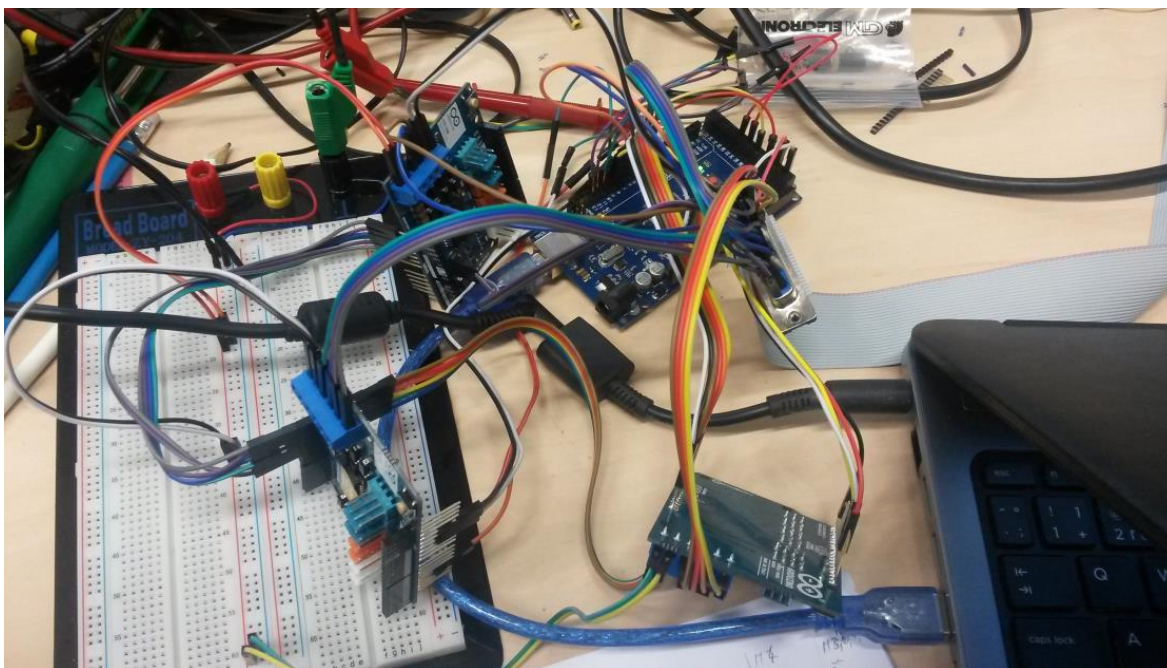
6.2. Řízení Teach-robotu

Veškeré potřebné dílčí úkoly byly dokončeny. Zbývá už pouze propojit všechny části dohromady a vytvořit tak finální řídicí program pro Teach-robot. Robot je poháněn šesti motory, je tedy nutné šestkrát replikovat program na ovládání motoru. Pro přehlednost jsem uzavřel celé schéma každého motoru do samostatného bloku. Každý z těchto bloků je třeba naladit pro potřeby konkrétního motoru. Na vstupu programu jsem umístil operátorem volené parametry, mezi které patří souřadnice koncového bodu, úhel s jakým uchopovací mechanismus přistupuje k tomuto bodu, úhel natočení zápěstní části a ovládání otevření a zavření úchopového mechanismu. Tyto údaje jsou přivedeny do bloku, který obsahuje program na výpočet požadovaných úhlů natočení jednotlivých segmentů. Výsledné úhly jsou vstupní hodnotou pro řízení příslušných motorů. Na výstupu je skutečná poloha všech šesti motorů. Celé schéma je zobrazeno na obrázku 40.



Obr. 40 – Řídicí program Teach-robotu

Program pro řízení je připravený, stačí jej nahrát na základní desku Arduino a připojit všechny motory. Základní deska je napájena přes USB kabel z PC. K ní jsem připojil tři Motor-Shieldy, které tvoří prostředníka mezi motory a základní deskou. Na každý ze Shieldů jsem přivedl 12V zdroj pro napájení motorů. Každým Shieldem lze napájet a řídit dva motory. Vše jsem propojil se sběrnici vedoucí k motorům Teach-robotu. Na obrázku 41 je ukázka zapojení při testování řídicího programu.



Obr. 41 – Propojení Hardware

Před spuštěním programu je nutné nastavit všechny vstupy a výstupy, aby odpovídaly správným pinům, na které jsou vedeny. Při testování funkčnosti řídicího programu robot reagoval na příkazy operátora, zadávané v prostředí Simulink. Tím byla ověřena funkčnost celého konceptu.

Posledním krokem bylo sestavení, prodrátování a spájení všech obvodů a uzavření veškeré řídicí elektroniky do krabičky, která má na vstupu připojení zdroje napájení pro motory a USB konektor pro připojení k PC. Na výstupu je konektor pro připojení sběrnice vedoucí k motorům a senzorům.

7. Závěr

V práci je popsána konstrukce Teach-robotu. Byl udělán rozbor elektronických obvodů u pohonu robotu a připravena nová elektronika potřebná pro řízení motorů. Byly analyzovány pohybové možnosti robotu a sestaveny kinematické modely pro určení jeho polohy. Pomocí maticové metody byl vytvořen model dopředné kinematiky a pomocí analytické trigonometrické metody sestaven model pro inverzní kinematiku. Tento model byl využit pro napsání programu v Matlabu na výpočet úhlů mezi jednotlivými segmenty v závislosti na poloze koncového bodu. Do 3D grafů byl vykreslen pracovní prostor Teach-robotu. Pomocí programu Simulink bylo vyřešeno načítání dat ze senzorů umístěných na motorech a tím problematika určování polohy motorů. Dále byl sestaven program pro ovládání rychlosti a směru otáčení motorů. Spojením těchto částí byl vytvořen řídicí program na polohování jednoho motoru. Zkombinováním výpočetního bloku napsaného v Matlabu a řídicích bloků sestavených v Simulinku vznikl ovládací program pro polohování všech motorů současně a tedy pro řízení polohy Teach-robotu. S využitím nové elektroniky byla vytvořena sestava, která je po nahrání programu na základní desku schopna ovládat všechny motory. Řídicí program byl nejprve otestován na jednom zkušebním motoru a následně přímo na celém robotu. Všechny cíle práce tak byly splněny.

8. Literatura

- [1] SKAŘUPA, Jiří. *Průmyslové roboty a manipulátory*. Ostrava: Ediční středisko VŠB – TUO, 2007, ISBN 978-80-248-1522-0.
- [2] RUMÍŠEK, Pavel. *Automatizace*. Brno: VUT 2003. Učební texty, VUT Fakulta strojního inženýrství Technolgy, 1997.
- [3] ŠŤASTNÝ, Petr. *Příprava výukového robota*. Praha: ČVUT 2013. Bakalářská práce, ČVUT, Fakulta strojní, Ústav mechaniky, biomechaniky a mechatroniky.
- [4] VALÁŠEK, Michael. *Kinematika robotických systémů*. Praha: ČVUT 2011. Učební texty Centrum pro rozvoj výzkumu pokročilých řídicích a senzorických technologií
- [5] KERMIT, Sigmon, *MATLAB Primer*, Druhé vydání, Gainesville: University of Florida 1992 Department of Mathematics
- [6] ŠIROKÝ, Michal, *Základy práce s programem Simulink*, Západočeská univerzity v Plzni 2007, Fakulta aplikovaných věd, Katedra kybernetiky
- [7] ZBYŠEK, Voda, *Průvodce světem Arduina*, 2014
- [8] ARDUINO [online] 2016, <https://www.arduino.cc/>
- [9] ARDUINO MEGA 2560 DATASHEET [online] <https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf>