

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

ROBOTICKÝ MANIPULÁTOR PRO ŘEŠENÍ HANOJSKÝCH VĚŽÍ

THE ROBOTIC MANIPULATOR FOR THE HANOI TOWERS SOLVING

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

RADIM SPÁČIL

VEDOUCÍ PRÁCE
SUPERVISOR

ING. TOMÁŠ MARADA, PH.D

BRNO 2015

ZADÁNÍ ZÁVĚREČNÉ PRÁCE

(na místo tohoto listu vložte originál a nebo kopii zadání Vaš práce)

ABSTRAKT

Bakalářská práce se zabývá návrhem a realizací manipulátoru pro řešení matematického hlavolamu Hanojské věže. Cílem práce je zhotovit přípravek, který bude automaticky řešit výše zmíněný matematický hlavolam a bude sloužit Ústavu automatizace a informatiky k prezentačním účelům. Práce se skládá z volby vhodného mechanického řešení. Následně je vybrané řešení navrženo v programu Autodesk Inventor 2015. Pomocí 3D tiskárny jsou zhotoveny unikátní díly, které není možné zakoupit a celý přípravek je smontován a oživen pomocí mikrokontroléru Arduino Mega 2560

ABSTRACT

Main theme of the bachelor's thesis is to design and manufacture a manipulator which can solve mathematic issue called the Hanoi towers. Main goal is to make a machine, which will automatically solve the above mentioned mathematic issue and will server for institute of automation and informatics for presentational purposes. This thesis contains of choosing the proper mechanical solution. Chosen solution will be afterwards design in Autodesk Inventor 2015 programme. Unique components which cannot be bought will be made by using 3D printer and machine will be assembled and then control by micro controller Arduino Mega 2560.

KLÍČOVÁ SLOVA

Hanojské věže, Robot, Arduino.

KEYWORDS

Hanoi towers, Robot, Arduino.

PROHLÁŠENÍ O ORIGINALITĚ

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně dle pokynů vedoucího a s použitím uvedené odborné literatury.

V Brně, dne 12.5.2015

BIBLIOGRAFICKÁ CITACE

SPÁČIL, F. *Robotický manipulátor pro řešení Hanojských věží*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2015. XX s. Vedoucí bakalářské práce Ing. Tomáš Marada, Ph.D.

PODĚKOVÁNÍ

Tímto chci poděkovat vedoucímu mé bakalářské práce Ing. Tomášovi Maradovi Ph.D., za vedení, které mi poskytnul v průběhu práce.

Obsah:

Zadání závěrečné práce

Abstrakt

Prohlášení o originalitě

Poděkování

1	Úvod	13
2	Motivace	15
3	Hanojské věže	17
3.1	Princip.....	17
3.2	Algoritmus řešení	18
4	Návrh manipulátoru	21
4.1	Příklady použitých manipulátorů.....	21
4.1.1	Univerzální manipulátory používající modelářské servomechanizmy	21
4.1.2	Univerzální manipulátory	22
4.1.3	Speciální manipulátory pro řešení úlohy Hanojských věží	22
4.2	Výběr vhodné koncepce a materiálu	23
4.3	Inventor 2015.....	24
4.4	Návrh manipulátoru	28
5	Řídící elektronika	33
5.1	Arduino obecně.....	33
5.2	Hardware	33
5.2.1	Arduino Mega 2560	34
5.2.2	RAMPS 1.4	34
5.2.3	Pololu DRV8825	35
5.2.4	Sestavení elektroniky	36
6	Realizace	37
6.1	Postup kompletace manipulátoru.....	37
6.2	Řídící Software	39
6.2.1	Arduino IDE	39
6.2.2	Ovládací program	40
7	Závěr	45
	Seznam použité literatury	
	Seznam příloh	

1 ÚVOD

Hanojské věže jsou matematický hlavolam, který je zařazen do skupiny hlavolamů, řešitelných pomocí postupných kroků. Jako takové jsou ideálním případem pro řešení pomocí robotického manipulátoru.

Cílem této bakalářské práce je sestavit přípravek, který bude zábavnou a interaktivní formou demonstrovat využití, koncepci a jednotlivé prvky robotického manipulátoru. Jedna z možností uplatnění je při dnech otevřených dveří na Fakultě strojního inženýrství VUT Brno.

Pokud například k úloze, kterou řeší robot, přidáme i klasickou dřevěnou variantu Hanojských věží, můžou si případní uchazeči o studium poměřit s robotem síly a zjistit, kdo bude rychlejší.

2 MOTIVACE

Jak již bylo zmíněno v úvodu, cílem této bakalářské práce je realizace robotického manipulátoru pro řešení úlohy Hanojských věží.

Mým cílem však je seznámení s možnostmi robotických manipulátorů v obecné rovině. Vzhledem k tomu, že v podstatě jakékoliv odvětví průmyslové výroby se dnes bez robotů neobejde a jejich využití je jak při zrychlení a zefektivnění výroby, tak při pracovních činnostech v prostředí pro člověka nebezpečném, beru téma bakalářské práce jako jedinečnou možnost pro seznámení se s jejich problematikou. Možnost jak pochopit jednotlivé fáze návrhu a vývoje robotického manipulátoru od samotného návrhu vhodného dizajnu pro zadaný úkol, přes konstrukční návrh v 3D modelačním programu, po realizaci řídicí elektroniky včetně příslušného software.

Všechny znalosti získané v průběhu bakalářské práce lze následně zúročit jak v zaměstnání, tak při tvorbě soukromých projektů, jako je například 3D tiskárna.

3 HANOJSKÉ VĚŽE

Dle legendy je v Indii chrám se třemi věžemi. Jsou to Věž zrození, Věž života a Věž zkázy. V každé věži je kůl, na kterém je navlečeno několik zlatých disků. Dohromady je ve věžích disků 64. Platí pravidlo, že lze umístit jen menší disk na větší, ne naopak. Poskládané disky na každém kůlu tedy vypadají jako kužel.

Při stvoření chrámu bylo všech 64 disků navlečeno na kůl ve Věži stvoření. Následně mniši, kteří chrám obývají, každý den přenesou jen jeden disk tak, aby co nejdříve dostaly všechny disky z kůlu ve Věži zrození na kůl ve Věži zkázy. Pokud tuto rutinu přeruší, zastaví se tok života. Na konci legenda říká, že až mniši všechny disky přesunou, nastane konec světa.

Ve skutečnosti hlavolam vymyslel francouzský matematik Édouard Lucas v roce 1883 a uveřejnil jej pod pseudonymem N. Clause de Siam. Tento pseudonym je anagram Lucasova jména ve tvaru Lucas d'Amiens, tedy Lucas z Amiensu. [1]

Hlavolam však neslouží jen jako hříčka pro zkrácení volné chvíle dětí a matematiků. Našel si svou cestu i do světa programátorů, kde se používá k vysvětlení rekurzivních řešení a v neposlední řadě jej používají psychologové pro zkoumání funkcí lidského mozku.

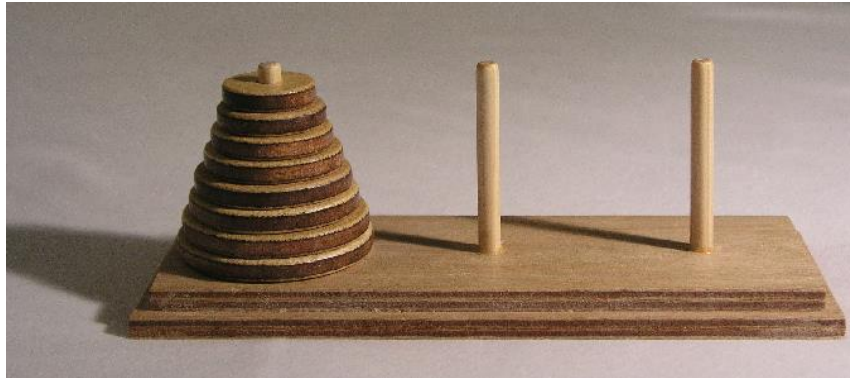
3.1 *Princip*

Už v předchozí podkapitole byl zmíněn základní princip fungování Hanojských věží. Pro jednoznačnost a jednodušší pochopení tohoto principu si jej zopakujeme již bez příběhu.

Hanojské věže se skládají z tří kůlů. Na začátku hry je na jednom kůlu navlečeno několik různě velkých disků. Největší disk je vespod, na něm je položen menší disk, na něm opět menší disk a tak se pokračuje, až je navrch vzniklé věže položen nejmenší z disků (viz Obr. 1).

Úkolem hlavolamu je přeskládat disky na jednu z prázdných věží a dodržet při tom následující pravidla:

- Za jeden tah lze přemístit jen jeden disk
- Tah je definován zvednutím disku z vrcholu jedné věže a jeho položení na vrchol jiné věže
- Je povoleno položit menší disk na větší, opačná situace je zakázána [2]



Obr. 1 Hanojské věže

3.2 Algoritmus řešení

Při psaní programu pro řešení úlohy hanojských věží je nejlepší cestou zvolit rekurzivní způsob řešení. Dejme tomu, že chceme přemístit n disků z věže jedna na věž dva. Abychom toto mohli udělat, musíme zákonitě přesunout největší disk z věže jedna na věž dva. K tomu potřebujeme, aby zbylých $n-1$ disků bylo umístěno na věži tři. Věž dva tedy bude volná a lze na ni přesunout největší disk z věže jedna. Následně můžeme přesunout $n-1$ disků z věže tři na věž dva. Tento přesun vykonáme pomocí rekurzivního volání tohoto algoritmu pro $n-1$ disků. Největší disk můžeme zanedbat, protože na něj lze položit jakýkoliv další disk.

```
void hanoj(int n, char pocatecni, char cilova, char odkladaci)
{
    if (n>1)
    {
        hanoj(n - 1, pocatecni, odkladaci, cilova);

        printf("\n Presun disk z veze %c na vez %c", pocatecni, cilova);

        if (n>1)
        {
            hanoj(n - 1, odkladaci, cilova, pocatecni);
        }
    }
}
```

Obr. 2 Funkce hanoj napsaná v C++ [2]

Algoritmus řešení psaný v jazyce C++ lze vidět na obr. 2. Jak bylo řečeno výše, nejdříve pomocí rekurzivního volání tohoto algoritmu přesuneme $n-1$ disků z věže jedna na věž tři. O to se stará první podmínka `if (n>1)`. Poté se přesune největší disk z první věže na věž druhou. A nakonec se pomocí druhé podmínky `if (n>1)` a rekurzivního volání tohoto algoritmu přesune $n-1$ disků z věže tři na věž dva.

4 NÁVRH MANIPULÁTORU

Před zahájením návrhu samotného manipulátoru proběhlo seznámení s již realizovanými roboty pro řešení úlohy Hanojských věží. Informace se získávaly z Internetu pomocí vyhledávače Google a ze serveru Youtube.

4.1 Příklady použitých manipulátorů

Na základě informací dle úvodu kapitoly 4, lze roboty klasifikovat do tří skupin, které jsou uvedeny níže.

4.1.1 Univerzální manipulátory používající modelářské servomechanizmy

Tato skupina obsahuje řadu opravdu unikátních manipulátorů. Ovšem nejčastěji se vyskytují dva koncepty manipulátoru. První můžeme vidět na obr. 3a. Jedná se o koncept robotického ramene založený na více stupních volnosti. [3] Druhý koncept je k vidění na obr. 3b a jedná se o manipulátor s dvěma stupni volnosti, který přenáší disky pomocí rotačního pohybu. [4]

Oba tyto manipulátory spojuje jak nepříliš estetická stránka, tak velká nepřesnost při umístění disků na kůly. Tato nepřesnost je způsobená provedením a použitím modelářských servomechanizmů. I v případě lepšího zpracování manipulátoru zde stále zůstane problém v podobě použitých modelářských servomechanizmů.

Tyto servomechanizmy jsou vyrobeny z plastu a při větší zátěži nemusí dosáhnout vždy stejné polohy nebo ji nedosáhnou bez překmitu. Se vzrůstajícím počtem stupňů volnosti a počtem servomechanizmů se situace ještě zhoršuje. Dále u modelářských servomechanizmů není možné určovat rychlost pohybu.

Nutnost vypořádat se s těmito problémy v důsledku vede k prodloužení jednotlivých tahů. Z těchto důvodů nebyla tato koncepce použita.



Obr. 3 a) Rob. rameno s více stupni volnosti; b) Manipulátor se 2 stupni volnosti

4.1.2 Univerzální manipulátory

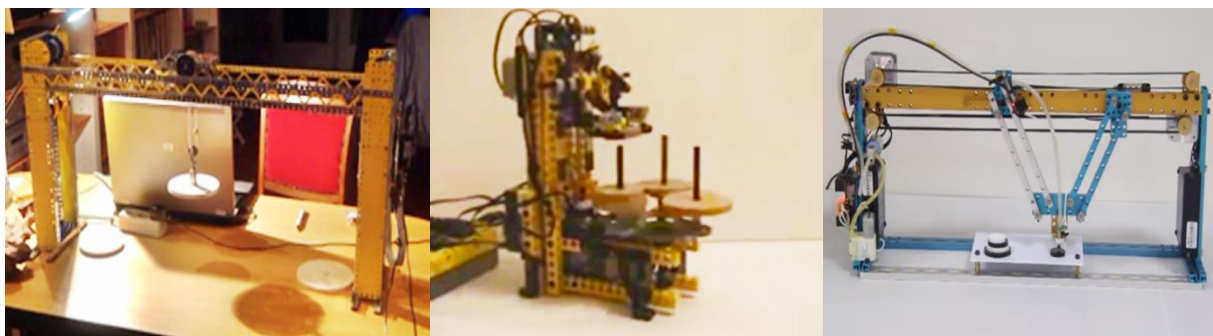
Tuto kategorii vyplňují profesionální průmyslové manipulátory určené k činnostem u výrobních linek (Obr. 4). Tyto manipulátory nechal výrobce, potažmo zapálený vývojář, naprogramovat pro řešení úlohy Hanojských věží. Přesnost je na zcela jiné úrovni než v případě první kategorie a výrazného zlepšení se dočkala i doba potřebná pro provedení jednoho tahu. Naneštěstí příslušný průmyslový manipulátor na ústavu k dispozici nemáme, tak bylo nutné i od této kategorie upustit.



Obr. 4 Průmyslový manipulátor

4.1.3 Speciální manipulátory pro řešení úlohy Hanojských věží

Do této kategorie lze zařadit veškeré manipulátory, které jsou stavěny s jediným záměrem, a to vyřešit úlohu Hanojských věží. Pro jiné úkoly se tyto manipulátory většinou nehodí. Dalším specifikem této kategorie je použití krokových motorů namísto modelářských servomechanizmů. Opět lze nalézt celou řadu kreativních řešení. Jako příklad lze uvést jeřábovou konstrukci, viz obr 5a. [5] Ještě zajímavěji působí rotační koncept na obr. 5b [6], kdy nedochází k přesunu jednotlivých disků mezi stacionárními věžemi, ale pouze k zvednutí disku a pomocí rotačního pohybu samotných věží k volbě cílové věže.



Obr. 5 a) Jeřábová konstrukce; 5b) Rotační koncept; 5c) Rovinný paralelní mechanismus

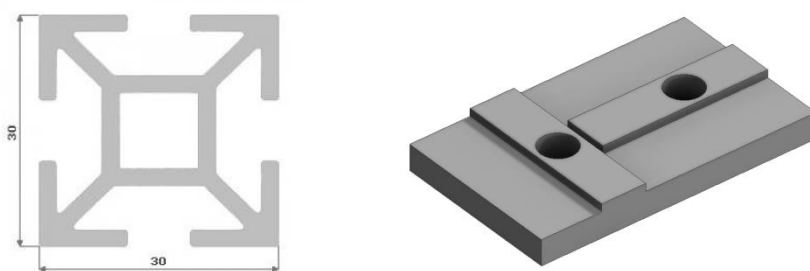
V této kategorii se nalézá i systém zobrazený na obr. 5c. Jedná se o rovinný paralelní mechanismus, který vykazuje jak přesnost, tak rychlost. [7] Na rozdíl od systému na obrázku 7 je i přehledný.

Tento systém vychází z dvojice vozíků pohybujících se po společné ose. Na každý z vozíků je uchycena dvojice souběžných ramen, které jsou na opačném konci přichyceny k podavači disků. Pokud se vozíky pohybují od sebe, dochází k zvedání podavače, pokud k sobě, tak podavač klesá. Při společném pohybu vozíků na jednu stranu dochází k rovnoměrnému přímočarému pohybu podavače na stranu, jako se pohybují vozíky. Zdvojení ramen slouží k stabilizaci podavače ve vodorovné poloze. Pro návrh, kterým se zabývají další podkapitoly, se vycházelo z tohoto typu manipulátoru.

4.2 Výběr vhodné koncepce a materiálu

Příklad systému pro řešení Hanojských věží a způsob, jakým lze zrealizovat, byl zobrazen v předchozí podkapitole 4.1.3 *Speciální manipulátory pro řešení úlohy Hanojských věží* na obr. 5c. Při návrhu vlastní realizace posloužil jako vzor koncept 3D tiskárny Rebel II dle projektu rebelove.org. Na základě tohoto konceptu byl zvolen Stavební systém Komby od společnosti A + A Pardubice spol. s r.o. (www.ehlinik.cz). Z tohoto systému jsou použity hliníkový profil Kombi-stojka, zobrazené v řezu na obr. 6a.

Velkou výhodou při návrhu a realizaci konceptu byla možnost využití 3D tisku konkrétně pomocí výše zmíněného modelu 3D tiskárny Rebel II. Za pomoci této tiskárny byly vyrobeny všechny unikátní části manipulátoru včetně spojek zobrazených na obr 6b. Tyto spojky následně slouží ke spojení jednotlivých hliníkových profilů a vytvoření celého rámu manipulátoru, na který se poté připevňovaly další prvky manipulátoru.



Obr. 6 a) řez Kombi-stojka; b) Spojka

Tato podkapitola slouží jako úvod k návrhu manipulátoru pro řešení úlohy Hanojských věží a jsou v ní představeny dva prvky, které slouží ke konstrukci rámu. Na těchto prvcích bude v dalších podkapitolách vysvětlena práce se softwarem pro 3D modelování.

Po základním vysvětlení fungování a práce s tímto softwarem přistoupíme k samotnému rozboru návrhu manipulátoru.

4.3 Inventor 2015

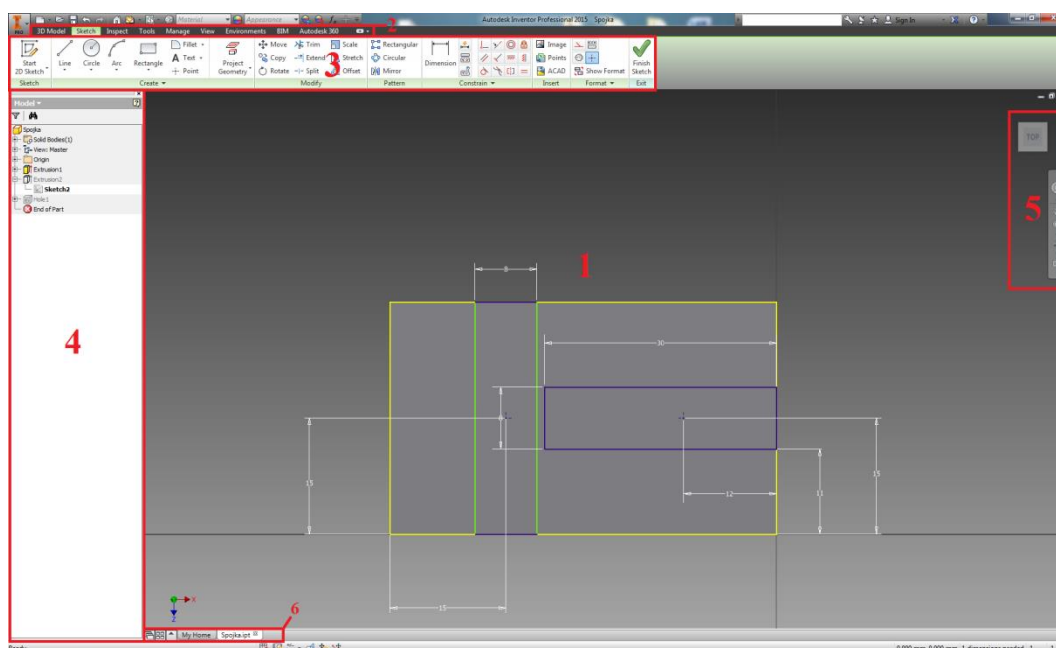
Před probíráním návrhu manipulátoru je vhodné alespoň ve stručnosti vysvětlit princip a používání softwaru, v kterém návrh vznikal. Jedná se o software od společnosti Autodesk, Inc. pod jménem Autodesk Inventor Profesional 2015.

Vyjma rychlého a efektivního návrhu 3D součástí se Inventor vyznačuje také možností použití vazeb mezi jednotlivými komponentami v sestavě. Tyto vazby následně umožňují simulovat vzájemné působení jednotlivých dílů a návrhář má tak k dispozici kompletní 3D model i s možností ověření chování v simulovaném provozu.

Před popisem rozhraní programu je vhodné vysvětlit postup tvorby 3D sestavy. Začíná se u sketche, což je návrh 2D výkresu. Následně se 2D výkres pomocí funkce Extrude „vytáhne“ do 3D modelu součástky. Jakmile je takto vytvořeno více 3D modelů, můžeme je pomocí funkce Contrain (vazba) spojovat do funkční 3D sestavy. Celý postup si probereme detailněji v dalších podkapitolách.

Rozhraní

Na obr. 7 je vzhled rozhraní programu Inventor 2015 s číselným označením jednotlivých sekcí. Tyto sekce se mohou lišit s ohledem na to, jestli navrhujeme 2D sketch, 3D model nebo vytváříme 3D sestavu. Na tomto obrázku je znázorněn stav při návrhu 2D sketche. Nyní budou rozebrány jednotlivé ovládací prvky programu. Pro přehlednost rozdělené do 6 sekcí.



Obr. 7 Rozhraní programu Autodesk Inventor 2015

Sekce:

- 1) Pracovní plocha, na které lze vidět a editovat samotný návrh. Ve spodním levém rohu lze vidět směrovou růžici, která udává orientaci jednotlivých os. Šedý kříž je pak průsečík os X a Z.
- 2) Tato sekce slouží k přepínání panelů nástrojů. Pro každou část návrhu (2D sketch, 3D model, sestava) existuje vlastní panel nástrojů se specifickými nástroji, které lze použít jen v daném návrhu. Nástroje z panelu nástrojů v 2D sketchi tedy nepoužijeme při navrhování 3D modelu.
- 3) Panel nástrojů, v kterém jsou nástroje příslušící k jednotlivým částem návrhu jako 3D modelování, 2D sketch, atd.. Blíže se jednotlivými nástroji budeme zabývat v dalších podkapitolách.
- 4) Browser – neboli prohlížeč jednotlivých částí návrhu. Například 3D model se skládá z několika 2D sketchů, které jsou transformovány pomocí funkce Extrude. Přehled jednotlivých použitých sketchů je právě v této sekci. Jiný příklad lze uvést při skládání sestavy z jednotlivých 3D modelů. Graficky vidíme sestavu v sekci 1 a zde s ní i pracujeme. Přehled všech použitých 3D modelů v sestavě pak nalezneme v sekci 4.
- 5) Tato sekce slouží ke změně pohledů na 3D model, sestavu nebo 2D sketch. Pokud se člověk naučí používat klávesové zkratky v kombinaci s myší, není pro něj tato sekce prakticky potřebná. Kombinace zkratk a myši je navíc o dost rychlejší. Následně v této sekci zůstane jen jeden potřebný nástroj, a to 3D kostka, která nám říká, z jakého pohledu se na 3D model díváme (z vrchu, zleva, ze spodu atd...)
- 6) Stejně jako máme v internetovém prohlížeči otevřeno více oken a můžeme se mezi nimi přepínat, tak můžeme mít v Inventoru 2015 otevřeno více projektů a přepínat se mezi nimi. Sekce 6 nám ukazuje, které projekty jsou otevřené a umožňuje nám jejich přepínání.

Následně budou probrány jednotlivé fáze návrhu, jsou jimi 2D Sketch, 3D model a celá sestava. Jednotlivé fáze nebudou probírány zevrubně, jelikož jsou do značné míry rozsáhlé a není cílem této práce zaměřit se jen na Inventor 2015.

2D sketch

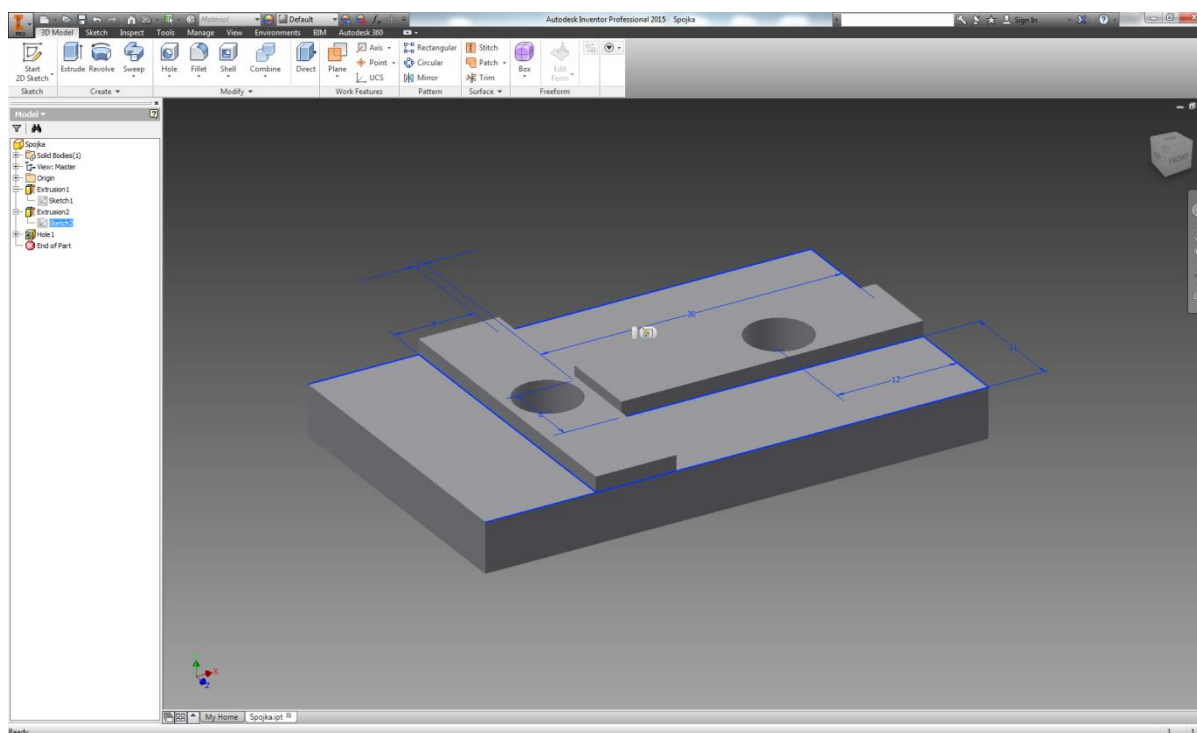
Na předchozím obrázku 11 je zobrazena první fáze návrhu, na které je 2D sketch spojky hliníkových profilů. V panelu nástrojů stojí za zmínku sekce Create, kde jsou nástroje na vytvoření jednoduchých 2D objektů jako je čára, čtverec, kruh atd. Dále je možné v sekci Modify jednotlivé nakreslené obrazce editovat (dělit, spojovat atd...). Další důležitou sekcí je Contrain, v které se nachází kótování a vazbení mezi jednotlivými obrazci. Všechny obrazce je potřeba zakótovat nebo zavazbit a jednoznačně tak určit jejich polohu. Pro vysvětlení pojmu vazbení si můžete představit bod, kterému pomocí vazby určíme, že je přesně v polovině mezi dvěma přímkami. Pokud má obrazec jednoznačně zadanou polohu, má modrou barvu, pokud ne, je jeho barva zelená.

3D model

Modelování 3D prvků primárně vychází z 2D sketche. Na obr 8 je v 3D modelu znázorněn 2D sketch, z kterého pomocí funkce Extrude vznikly dvě horní pera pro přidržení spojky v hliníkových profilech.

V panelu nástrojů se v sekci Create nachází již zmiňovaná funkce Extrude. Dále jsou podstatné funkce Hole a Fillet v sekci Modify. Funkce Hole slouží k tvorbě děr, a to jak zapuštěných v materiálu, tak průchozích skrz materiál. Funkcí Fillet je možné zaoblovat hrany materiálu. Tato funkce slouží jak k srážení hran, tak k získání radiusu v rozích 3D modelu.

V levé části obrazovky (na obrázku 7 je to sekce 4) se nachází okno Exploreru. V tomto okně je přehled jednotlivých prvků, ze kterých se 3D model skládá. U spojky na obrázku 12 lze vidět dvě Extrusion (Objekt vzniklý z 2D sketche použitím funkce Extrude) a sketche, z kterých vychází. Dále je zde Hole, tedy objekt zastupující dvě shodné díry v 3D modelu. Všechny tyto objekty je možné modifikovat.

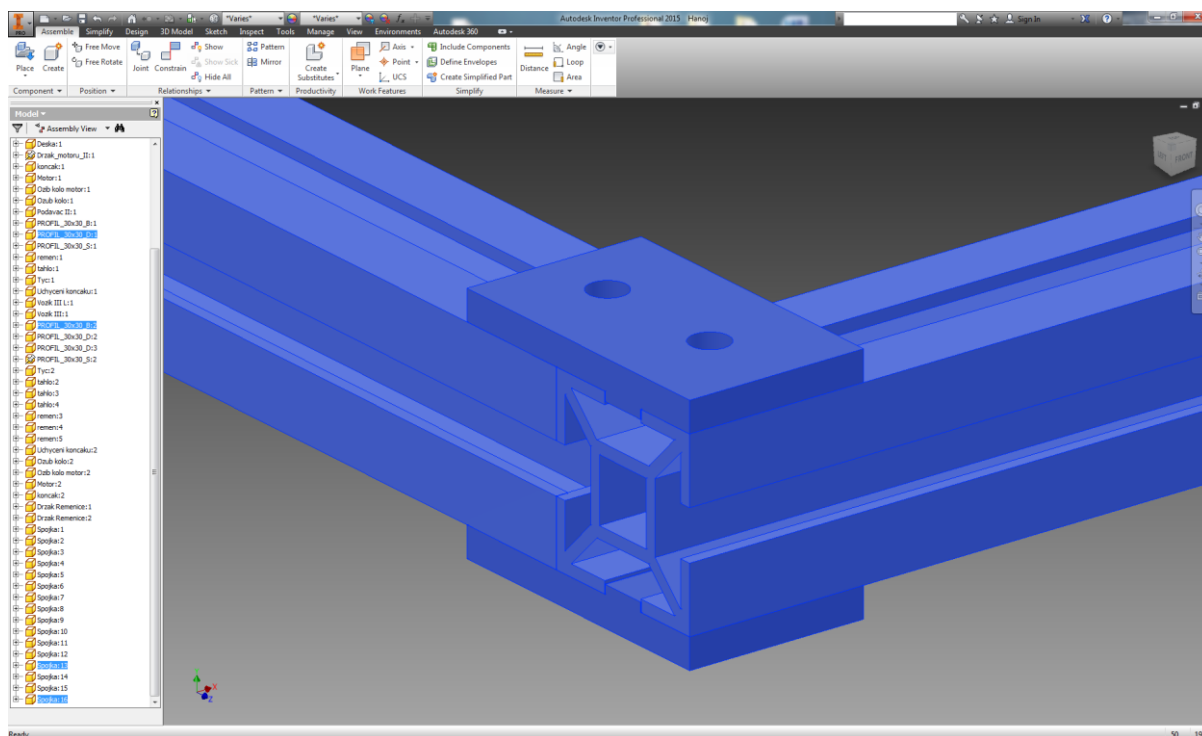


Obr 8 Rozhraní při návrhu 3D modelu

Sestava


Ve slovníku Inventrou se sestavě říká Assembly, a stejně tak se jmenuje i její panel nástrojů. Na obr 9 je zobrazeno spojení dvou hliníkových profilů pomocí dvojice výše uvedených spojek. Modrá barva prvků není náhodná a signalizuje, že prvky byly označeny (jeden prvek se vybere pomocí kliknutí levého tlačítka myši, více prvků pak pomocí Ctrl+klik). Takto označené prvky se modře zvýrazní i v Exploreru (na obrázku 11 sekce 4). Toto usnadňuje procházení a následnou editaci prvků.


V Panelu nástrojů je dobré upozornit na funkci Contrain, což jsou již zmiňované vazby, a také na funkci Distance, která umožňuje zjistit vzdálenost mezi jednotlivými částmi 3D modelů.




Obr 9 Rozhraní při návrhu sestavy

Při návrhu sestavy manipulátoru pro řešení úlohy Hanojských věží byly použity nejčastěji tyto vazby z funkce Contrain:

Mate – jedná se o vazbu volně přeloženou jako „kamarád“. Při použití této vazby se označí dvojice ploch (může být i dvojice os) na různých 3D modelech. Ty pak k sobě „přilnou“. Tato vazba ubírá jeden stupeň volnosti. Jako příklad lze uvést horní spojku, kde pomocí této vazby řekneme, že spodní plocha spojky a horní plocha hliníkového profilu mají být „kamarády“. Tím pádem na sebe dosednou .

Flush – jedná se o podskupinu vazby Mate. Opět označujeme dvě plochy, jen s tím rozdílem, že nedosedá jedna na druhou, ale zarovnají se do stejné roviny .

Insert – jedná se o rotační vazbu, kterou lze použít jen na tělesa s rotační částí, jako je díra nebo hřídel. Opět se vybírá dvojice těles. Tato vazba umístí tělesa na společnou osu a současně mezi nimi naváže kontakt (podobně jako vazba Mate). Vazba odeberá tak dva stupně volnosti .

4.4 Návrh manipulátoru

Jak bylo zmíněno na začátku této kapitoly, část koncepce manipulátoru vychází z 3D tiskárny Rebel II. Například pro tvorbu rámu bylo použito hliníkových profilů, které se pospojují spojkami z ABS plastu vytisknutými na 3D tiskárně.

Pro určení rozměrů rámu bylo nutné určit velikost ramen, které spojují vozíky s podavačem. Následně se určil pracovní prostor pro pohyb vozíků. K tomu účelu posloužil program MATLAB R2013.

Pro vykreslení poloh ramen byla použita rovnice 1. Pomocí cyklu for se v každém kroku vykresluje rozdílná pozice vozíků odstupňovaná o 1 cm. V každém kroku cyklu je bod na ose x reprezentující vozík spojen s podavačem disků.

$$h=(\sin(\arccos(i/r)))\cdot r \quad (1)$$

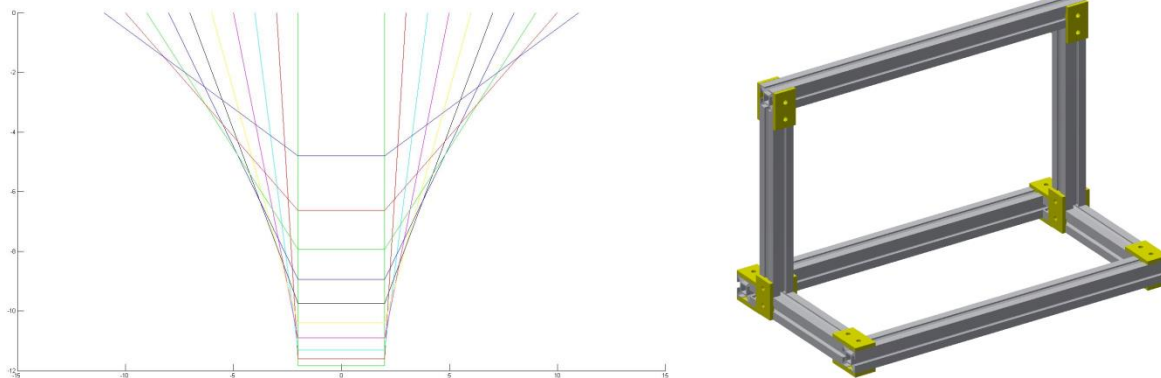
h...vzdálenost podavače od osy x

i...vzdálenost vozíku od počátku osy x

r...délka ramene

Při plánované výšce věži 45mm a potřebné délce zdvihu podavače 60mm vychází jako ideální délka ramen 120mm. Na obr. 10a je zobrazeno výsledné vykreslení. Z tohoto obrázku je patrné, že čím dále jsou vozíky od sebe, tím větší bude zdvih podavače na 10mm dráhy vozíků. Tato skutečnost hraje roli pro rychlost samotného manipulátoru. Pokud se vozíky budou pohybovat blízko u sebe, musí urazit značnou dráhu pro daný zdvih. Pokud se však budou pohybovat v krajních polohách, stačí jim dráha daleko kratší, kterou logicky urazí za kratší čas. Maximální vzdálenost vozíků je tak 280mm. 120mm má každé rameno a 40mm má podavač. Technicky však není rozumné chodit do této maximální vzdálenosti, proto se dál počítá jen s 270mm. Následně je třeba započítat vzdálenost mezi jednotlivými věžemi, která byla stanovena na 35mm.

Dále místo pro koncové spínače a pro části vozíků, které do tohoto okamžiku byly reprezentovány jen bodem. Ve výsledku je stanoven vnitřní rozměr rámu na 400mm a celkový rozměr nejdelšího hliníkového nosníku 460mm (2 x 30mm pro položení na boční stojny). Celý rám je pak zobrazen na obr. 10b.

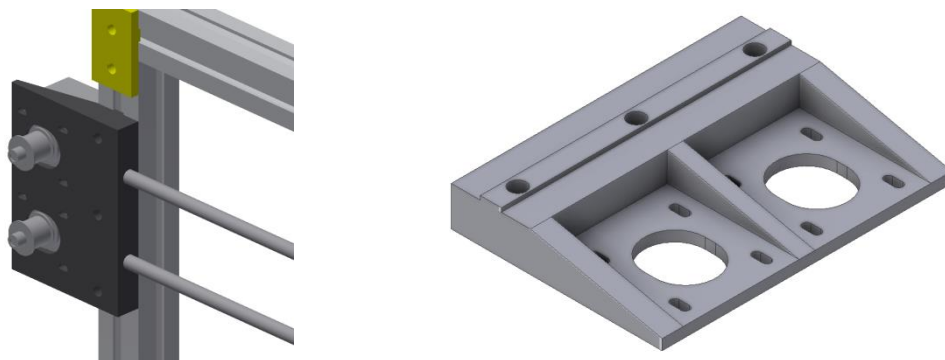


Obr. 10 a) Rozsah ramen $d=120\text{mm}$; b) Navržený rám

Dále bylo nutné navrhnout systém, pomocí kterého se budou vozíky pohybovat. Zde se opět čerpalo z konstrukce 3D tiskárny Rebel II, kde jsou jako vodící prvky pro extruder použity tyče, po kterých umožňují pohyb lineární ložiska. Pro tento manipulátor byla zvolena dvojice 8mm tyčí.

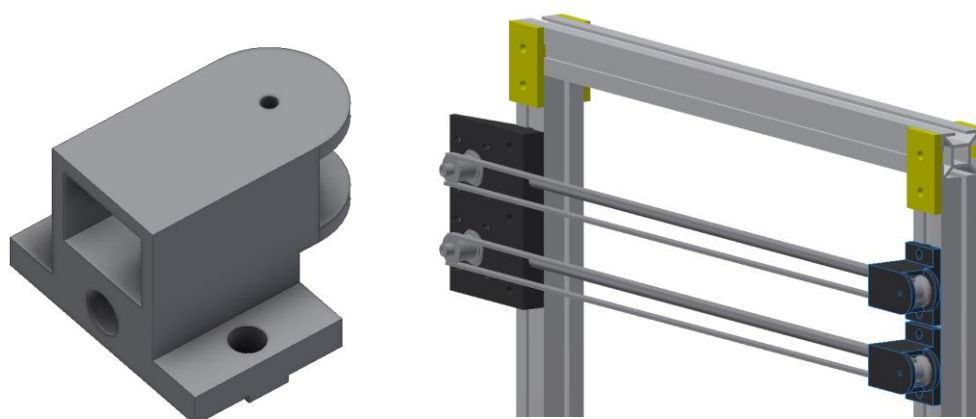
Pohonnou jednotkou jsou v tomto případě krokové motory. Převod otáčení výstupní hřídele motoru na lineární pohyb vozíků je zprostředkován pomocí řemenice a ozubeného řemene.

Tyče i motory je nutné přichytit k rámu manipulátoru. Ideální je navrhnout takovou součástku, která bude plnit obě funkce současně. Na obrázku 11a je součástka zachycena již osazená na rámu včetně připevněných tyčí a dvojice motorů, které mají na hřídelích nasazeny řemenice. Na ob. 11b je součástka zachycená z druhé strany. Na druhém pohledu lze vidět pero pro zafixování držáku v hliníkovém profilu. Dále je zde dvojice otvorů pro osazení krokových motorů, které jsou oddělené žebry pro zvýšení pevnosti. Povšimněte si, že díry v oblasti otvorů pro motory nejsou kulaté, ale jsou oválné. Toto řešení umožní motoru stranový pohyb v rámci pár milimetrů, což bude použito pro napínání řemenic.



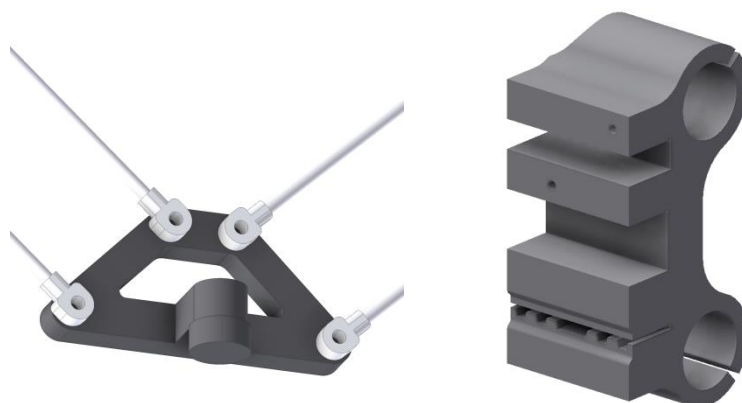
Obr. 11 a) Držák motorů na rámu; b) Samostatná držák motorů

Na druhé straně rámu se zákonitě musí nacházet další součástka pro uchycení tyčí a dalších řemenic. V tomto případě však není použita jediná součástka, ale je použito dvojice totožných součástek. Takové řešení bylo zvoleno s ohledem na možnost nastavení souběžnosti vodících tyčí. Na obr. 12a je samotná součástka. V dolní části součástky je opět pero pro fixaci v hliníkovém profilu, směrem nahoru následuje otvor pro vodící tyč. V horní části je pak průchozí kaplička pro upevnění řemenice. Řemenice je jednoduchá, bez ozubení, a také vytištěna na 3D tiskárně. Za poznámku stojí, že v každé řemenici je dvojice ložisek 623 2Z. Na obr. 12b je pak celková situace i s nasazeným řemenem. Držáky a řemenice jsou zvýrazněny modrými obrysy.



Obr. 12 a) Držák vodících tyčí s kapličkou; b) Dvojice osazených držáků

Do kompletního návrhu chybí už jen vozíky, ramena a podavač. Na obr. 13a je znázorněn podavač v pohledu zespod, kdy je patrný i magnet pro zachycení disků. Na obr. 13b je uveden vozík. Prostorově se jedná o nejkomplicovanější součástku. Do dvojice velkých průchozích děr na spodní straně vozíku se vkládají lineární ložiska pro posun na vodících tyčích. V dolní vystouplé části vozíku je ozubení, do kterého se vkládá řemen. Řemen není spojitý, a tak toto ozubení slouží pro jeho spojení. Z tohoto důvodu byl také vynechán prostřední zub. V blízkosti ozubení je dvojice malých děr sloužících pro upevnění ramen manipulátoru.



Obr. 13 a) Podavač ; b) Vozík

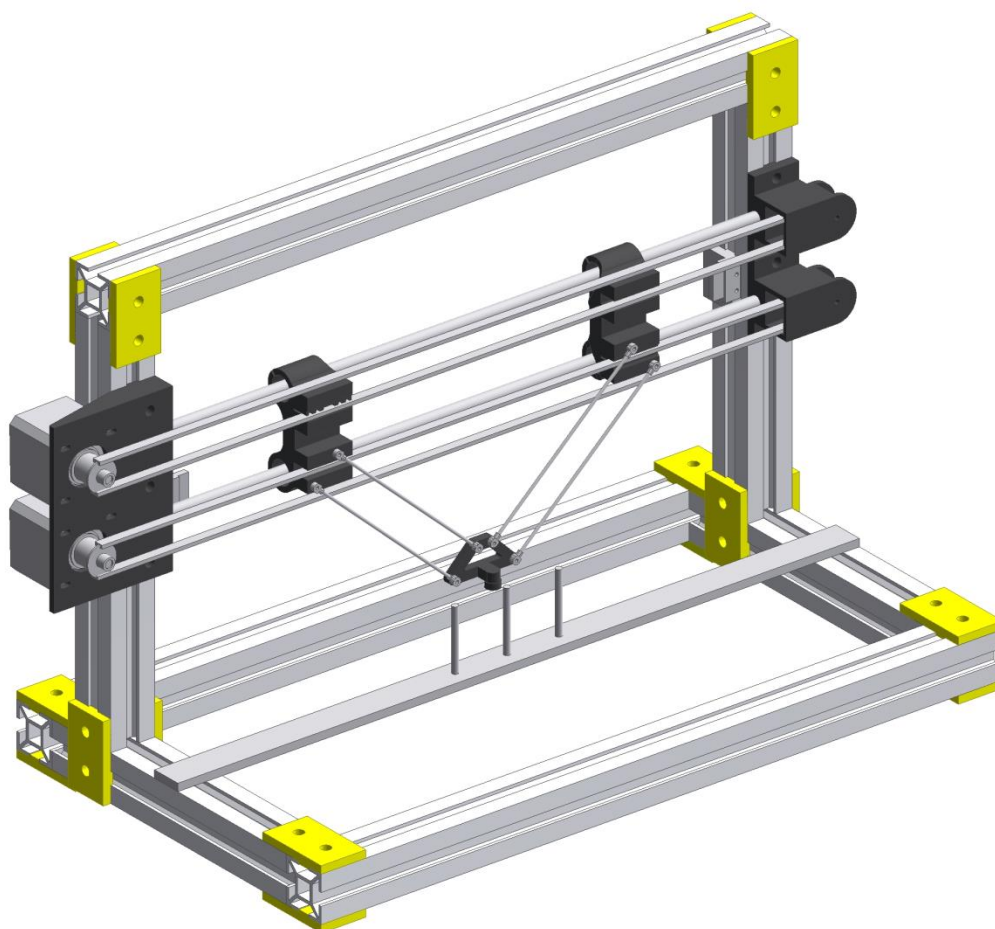
Ramena manipulátoru jsou realizována pomocí kulového čepu V1, průměr 5mm, M2/2 dlouhý a 2mm ocelového drátu.

Celá sestava je následně znázorněna na obr. 14. Je zde patrné spojení vozíků, ramen a podavače disků. V dolní části je pro ilustraci uvedena jedna z možných podob Hanojských věží. Upevnění do hliníkových profilů se realizuje pomocí šroubů M6 a pojistných matic. Motory se uchycují pomocí šroubů M3. Stejnými šrouby, jen s jinou délkou, se uchycují řemenice na samostatných držácích tyčí a řemenic v pravé části rámu. Čepy ramen se uchycují pomocí šroubů M2.

Pokud se na obrázek podíváte pozorněji, uvidíte, že levý vozík má ozubení pro napojení na řemen v horní části a je tedy ovládán horním krokovým motorem. Pravý vozík má pak ozubení v dolní části a je ovládán spodním krokovým motorem.

Na pravé boční stojně lze v oblasti mezi ozubenými řemeny vidět model koncových spínačů.

Pro manipulátor byl zvolen ozubený řemen standardu T5 o šířce 6mm a ozubená řemenice s 10 zuby typu 1F.



Obr 14 Celá sestava manipulátoru

5 ŘÍDÍCÍ ELEKTRONIKA

Jako vhodná řídicí elektronika byl zvolen modulární systém Arduino. Proč došlo k této volbě a v čem je tento systém výhodný pro robotický manipulátor řešící úlohu Hanojských věží, bude probráno v následujících podkapitolách.

5.1 *Arduino obecně*

Arduino je hardwarová open-source platforma sloužící jako vývojový kit pro začínající i pokročilé programátory. Celosvětově se těší velké přízni, a to hlavně díky široké škále levných přídatných modulů, které je možné dokoupit. Ve světě Arduina se těmto přídatným modulům říká shields (štíty).

Obliba Arduina je způsobená také open-source licencí, konkrétně Creative Commons Attribution Share-Alike 2.5. Na základě této licence autoři uveřejnili podrobná schémata zapojení a příslušnou dokumentaci. Hlavně však dali svolení k modifikaci těchto schémat a stavbě výsledného hardware. To vše zcela bez nároku na jakoukoliv finanční odměnu pro autory.

5.2 *Hardware*

Jádrem Arduina je mikrokontrolér z rodiny ATmega od norské firmy Atmel. Většina těchto mikrokontrolérů je postavena na architektuře AVR. Jsou to tedy 8 bitové mikroprocesory typu RISC s hardwarskou architekturou. Jedná se především o typy ATmega8, ATmega168 a ATmega328. Mikrokontroléry se liší hlavně velikostí RAM a FLASH paměti. Nejnovější Arduino DUE je již založeno na procesorech s 32-bitovou architekturou ARM Cortex-M3 také od norského Atmelu.

Každý mikrokontrolér má v sobě integrované některé z těchto periférií: časovače, sériové a paralelní porty, digitální vstupy a výstupy, A/D převodníky atd... Dále desky s mikrokontroléry obsahují dodatečné elektronické obvody, jako stabilizátory napětí, rozhraní pro komunikaci s počítačem, atd.. [8]

5.2.1 Arduino Mega 2560

Jedná se o jednu z největších vývojových desek v projekt Arduino. V tabulce 1 jsou uvedeny technické parametry této desky. [9] Následně je na obr 14 vyobrazena vývojová deska.



Obr 14 Arduino Mega 2560

Tabulka 1 parametry pro Arduino Mega 2560

Microcontroller	ATmega2560
perating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

K použití této desky vedla hlavně možnost osadit ji shieldem RAMPS 1.4 , který následně umožňuje připojení driverů pro ovládání krokových motorů.

5.2.2 RAMPS 1.4

Jde o shield primárně vyvinutý pro ovládání 3D tiskáren. Mezi jeho výhody patří implementace napájecích obvodů pro drivery krokových motorů a následně i patice pro připojení těchto driverů.

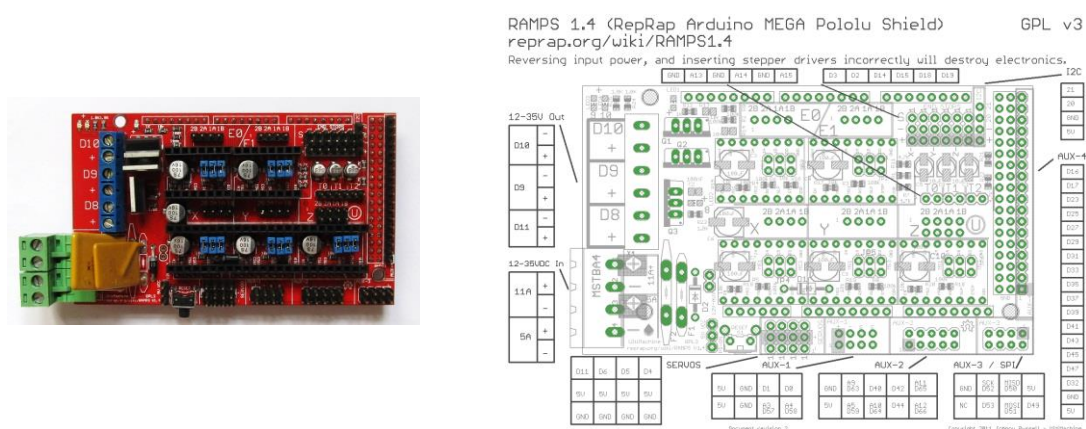
Z dalších vlastností shieldu je pro manipulátor důležitá implementace vstupních pinů

pro připojení Pullup koncových spínačů. Těmito spínači v neseprnutém stavu protéká proud. Při sepnutí spínače dojde k přerušení obvodu a zastavení toku proudu. Výhoda tohoto spínače je, že v případě poruchy se tváří jako sepnutý a snižuje se tak pravděpodobnost poškození celého zařízení.

Vyobrazení tohoto shieldu naleznete na obr. 15a, schéma zaměřené na rozložení pinů pak na obrázku 15b.

Jelikož je tento shield určený pro obsluhu 3D tiskáren, je tomuto faktu přizpůsoben i popis pinů. Na obrázku 22 tak lze vidět sekce označené X, Y a Z. Je zjevné, že označení je převzato z názvů os kartézské soustavy. V každé sekci jsou piny pro připojení driveru krokového motoru (viz následující podkapitola) a piny pro připojení samotného motoru. Dále jsou tu sekce E0 a E1 s piny pro připojení driverů a krokových motorů pro extrudery. V pravém horním rohu shieldu jsou pak piny pro připojení koncových spínačů (END STOPS). [10]

Vzhledem k možnosti vysokých proudových odběrů krokových motorů a tavných trysek 3D tiskárny má tento shield svorky pro připojení externího zdroje. Po připojení zdroje je přes shield napájeno i samotné Arduino.



Obr. 15 a) RAMPS 1.4; b) Schéma RAMPS 1.4 – mapa pinů

5.2.3 Pololu DRV8825

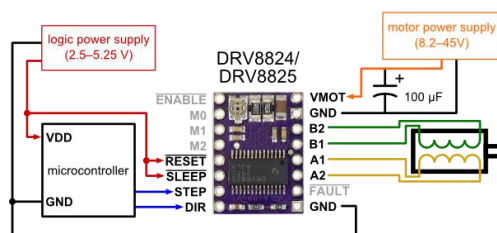
Pololu DRV8825 je driver sloužící k ovládání krokových motorů a lze jej osadit na RAMPS 1.4. Pololu je název společnosti, které vyrábí desku s elektronikou a řídicím čipem DRV8825, který je od společnosti Texas Instruments .

Tento driver slouží k ovládání bipolárních krokových motorů a zvládá mikrokrokování až do 1/32 kroku motoru, což umožňuje zvýšit přesnost krokového motoru až dvaatřicetkrát. V případě tohoto manipulátoru však dostačuje základní velikost kroku motoru. Výstupní proud může být až 2.2A a napájecí napětí až 45V. Napěťové úrovně logiky jsou pak 3.3V a 5V.

Driver se řídí pomocí přivádění signálů na jeho piny (Viz. Obr 16). Pro účely řízení manipulátoru jsou podstatné tyto: pin ENABLE povoluje provoz driveru, pin DIR určuje směr otáčení a pin STEP dává pokyn pro krok motoru.

Dále je na desce potenciometr pro nastavení proudu do krokového motoru.[11]

Pro účely řízení manipulátoru řešícího úlohu Hanojských věží lze pořídit i daleko levnější drivery pro Arduino (například Pololu A4988), nicméně DRV8825 byly vybrány s přihlédnutím k možnému použití i v jiných aplikacích.



Obr. 16 Schéma pinů na Pololu DRV8825

5.2.4 Sestavení elektroniky

Základem pro montáž řídicí elektroniky je vývojová deska Arduino Mega2560. Na tuto desku se pomocí pinů připevňuje shield RAMPS 1.4.

Protože je deska i shield od různých čínských výrobců, tak i přes deklarovanou kompatibilitu musela být nejdříve odstraněna část jednoho napájecího kontaktu na shieldu. Po tomto zásahu bylo možné shield osadit na Arduino.

Následně byla osazena dvojice driverů Pololu DRV8825 do pozic pro osy X a Y. Vedle těchto pozic jsou i piny pro připojení krokových motorů.

Jak již bylo řečeno v předchozí podkapitole, v pravém horním rohu shieldu jsou piny pro koncové spínače. Do těchto pinů se zapojují i spínače použité pro tento manipulátor. Konkrétně spínače s označením P-B172C.

Pro napájení celé sestavy lze použít libovolný zdroj s rozsahem stejnosměrného napětí 12-35V a příslušnou proudovou zatížitelností. Pro první pokusy s manipulátorem byl použit běžný počítačový ATX zdroj.

6 REALIZACE

Kapitola realizace manipulátoru se dělí do podkapitol zabývajících se vlastní kompletací manipulátoru z nakoupených a vytištěných dílů a následnou tvorbou řídicího software.

6.1 *Postup kompletace manipulátoru*

První krok při stavbě manipulátoru byla montáž rámu. Hliníkové profily ze Stavebního systému Komby byly objednány již v přesných rozměrech, proto je nebylo třeba jakkoliv upravovat. Spojky byly vytisknuty v předstihu, a tak po dodání hliníkových profilů mohl být rám zkompletován.

Při kompletaci se vyskytl jen drobný problém v podobě rozdílné výšky pojistných matic M6. Pravděpodobně prodejce dodal matice z různých výrobních sérií, a zatímco některé šly do rámu vložit bez sebemenšího problému, jiné bylo nutné mírně snížit. Pro potřebné snížení naštěstí stačilo ubrat nanejvýš desetinu milimetru a postačil smirkový papír drsnosti 80.

Po kompletaci rámu přišla řada na osazení následujících částí: kombinovaný držák motorů a tyčí, samotné tyče, vozíky a držáky tyčí s kapličkami pro řemenice. Všechny tyto části musely být nejdříve sesazeny a až následně uchyceny na rám manipulátoru.

Nejdřív se osazovaly tyče do držáku motorů. Aby tyče šly osadit, bylo potřeba výstružníkem protáhnout díry ve vytištěném držáku, jelikož při tisku došlo v krajních vrstvách děr k „vyboulení“.

Pro manipulátor byly zvoleny 8mm hlazené nerezové tyče. Před osazením se otestoval chod lineárních ložisek po tyčích a projevilo se mírné zadrhávání. Proto byly tyče před osazením ještě přešetřeny, což vedlo ke zlepšení chodu ložisek.

Následovalo osazení dvojice vozíků, do kterých byly vlepeny lineární ložiska. Pro vlepení bylo použito obyčejné „sekundové“ lepidlo.

Po osazení vozíků byly tyče zakončeny dvojicí držáků s kapličkami. Zde bylo opět nutné protáhnout díry pomocí výstružníku.

Takto zkompletovaný celek již bylo možné osadit na rám manipulátoru. Postup pro osazení byl následný:

1. Osazení a upevnění kombinovaného držáku motorů a tyčí k rámu pomocí 3 šroubů M6
2. Dvojice jezdců byla přesunuta zcela doprava na doraz k dvojici držáků tyčí s kapličkami. Tímto krokem byla zajištěna správná vzájemná vzdálenost držáků.
3. Uchycení dvojice držáků s kapličkami. Každý držák byl k rámu uchycen dvojicí šroubů M6.

Po uchycení tohoto celku byla k rámu připevněna ještě dvojice koncových dorazů, každý z jedné vnitřní strany rámu. Koncové dorazy byly umístěny ve vodorovné rovině s osami vodících tyčí, tak aby je vozík ve své krajní poloze vždy sepnul.

V tento okamžik již bylo možné osadit krokové motory. K držáku se každý připevnil pomocí čtyř šroubů M3 s pojistnými podložkami. Šrouby se nedotahovaly a s motory bylo možné pohybovat (viz kapitola.....). Po osazení motorů byly na jejich hřídele připevněny ozubené řemenice.

Následně se do vytištěných neozubených řemenic vložila ložiska a celý komplet se vložil do kapliček držáků tyčí na druhé straně manipulátoru a upevnil pomocí šroubů M3.

Po montáži těchto součástí bylo možné přistoupit k montáži ozubeného pásu. Ten se nejdříve protáhnul kapličkami s neozubenými řemenicemi. Následně se obtočil kolem ozubených řemenic na hřídelích motorů a zasunul se do připravených drážek ve vozíku.

Po spojení řemene pomocí vozíku se změnou polohy krokového motoru řemen napnul. Z tohoto důvodu se při osazování krokových motorů nedotahovaly šrouby určené k uchycení motoru k držáku. V tento okamžik se šrouby dotáhly.

Na řadu přišla montáž ramen a podavače. Dvoumilimetrový ocelový drát byl nastříhán na patřičnou délku, která se vypočítala na základě rozměrů použitých čepů a požadované vzdálenosti mezi osami čepů na rameni, která je 120mm. Následně se drát zasunul do čepů, kde držel i bez použití lepidla. Tímto způsobem se sestavila 4 ramena, která byla následně přišroubována k vozíkům a podavači.

Pro dočasné přichycení disků k držáku byl zvolen permanentní magnet. K uvolnění disku od magnetu je použit postup, kdy držák s magnetem najede svisle na vrchní část věže a následně pomocí lineárního pohybu do strany dojde k „utržení“ disku od magnetu. Disk zůstane na dané věži.

Posledním krokem před montáží elektroniky bylo vytvoření samotných Hanojských věží a k nim příslušných disků. Disky jsou vystřiženy z 0,5mm tlustého plechu o průměru 18-31mm. Pro zvětšení tloušťky jsou kovové disky podlepeny dvěma vrstvami tvrdého papíru. Disků bylo vytvořeno celkem 7.

Po sestavení mechanických částí manipulátoru přišla řada na elektroniku. Bližší informace k ní byly uvedeny již v předchozí kapitole. Zde je třeba uvést, že po zapojení elektroniky a napsání jednoduchého programu pro pohyb vozíků došlo k nastavení minimálního potřebného proudu do krokových motorů. Proud se nastavuje pomocí potenciometru na driveru krokového motoru – DRV8825. V případě tohoto manipulátoru byl nastaven proud 150mA, což je prakticky minimální proud, který lze na driveru nastavit.

6.2 Řídící Software

V této podkapitole bude nejprve probrán programovací software. Pak se přistoupí k popsání postupu, jakým vznikal samotný řídicí program pro manipulátor.

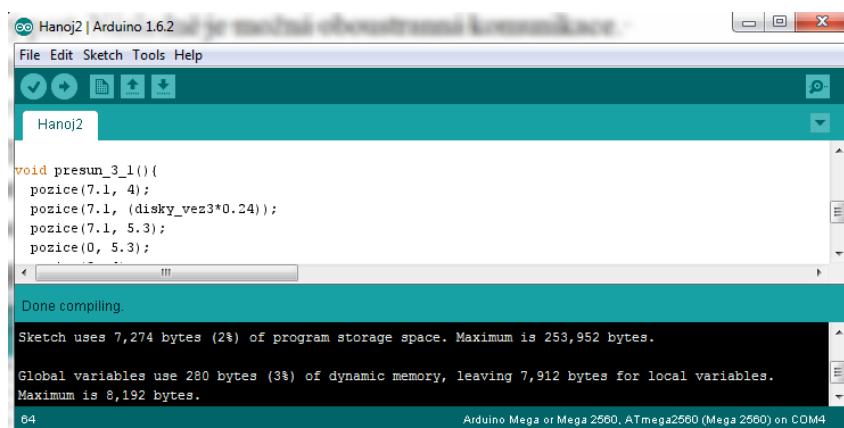
6.2.1 Arduino IDE

Jedná se o efektivní software sloužící k programování Arduin. Na obr. 17 je zobrazeno hlavní rozhraní pro tvorbu programu. Před zahájením samotného psaní programu je potřeba zvolit, pro jaký typ Arduina se bude program psát. Tato volba se provádí v sekci Tools->Board. Dále se volí komunikační port, ke kterému je Arduino připojené. Port lze zjistit pomocí Správce zařízení v sekci Porty (COM a LPT).

V levém horním rohu lze pod lištou s nabídkami File, Edit, Sketch, Tools a help nalézt dvojici kulatých tlačítek. První tlačítko, které obsahuje fajfku, slouží ke zkompilování programu. Pokud bude program správně zkompilován, zobrazí se ve spodní části okna hlášení Done Compiling a informace o výsledném zkompilovaném programu. Pokud při kompilaci dojde k chybě, zobrazí se hlášení Error Compiling a vypíše se typ chyby a řádek s jejím výskytem.

Druhé kulaté tlačítko, v kterém je šipka, slouží k uploadu programu do připojeného Arduina. Opět funguje princip s hlášeními o úspěchu nebo neúspěchu prováděné operace.

Pokud jsou v programu implementovány funkce pro komunikaci po sériovém portu, lze použít tlačítko s lupou, které je v pravém horním rohu aplikace. Toto tlačítko otevře konzoli pro komunikaci přes sériový port. Následně je možná oboustranná komunikace. Spuštěním této konzole dojde k restartu celého Arduina. Konzoli je možné využívat i jako zdroj stavů Arduina při ladění programu. V některých okamžicích není jasné v jakém stavu se Arduino nachází nebo proč došlo k poruše. V takovém případě lze do programu implementovat chybové hlášení a nechat je i s hodnotami proměnných posílat na sériový port a následně je pomocí konzole vypisovat přímo na obrazovku počítače.



Obr. 17 Arduino IDE

Všechny programy psané pro Arduino mají dvě předem dané části. První je část nazývaná se „setup“. V programu se volá „void setup() {obsah části }“. Tato část obsahuje takzvané nastavení programu a vykonává se jen jednou. Například se v ní nastavují typy a použití vstupně-výstupních pinů.

Druhá část se nazývá „loop“ a volá se pomocí „void loop(){obsah části }“. V této části je obsažené tělo samotného programu, který Arduino vykonává. Část se provádí stále dokola. Jedná se tedy o „nekonečný“ cyklus.

Deklarace proměnných a načtení případných knihoven se provádí jako první, tedy ještě před oběma výše zmíněnými částmi.

Také lze mimo tyto části definovat vlastní funkce, které jsou následně z těchto částí volány.

Když zde již byly zmíněny knihovny, je třeba uvést, že již v základní instalaci programu Arduino IDE je jich implementováno velké množství. Dále jsou součástí instalace i ukázkové programy. [12]

6.2.2 Ovládací program

Program pro řešení úlohy Hanojských věží pomocí robotického manipulátoru se skládá z následujících částí:

- Část s definicí a deklarácí proměnných a pinů
- Setup, kde jsou nastaveny potřebné proměnné a proveden inicializační program
- Loop, odkud jsou volány funkce
- Poslední část, kde jsou definovány funkce pro řízení manipulátoru

Na začátku samotného programu jsou za pomoci dokumentace k RAMPS 1.4 definovány názvy pro jednotlivé piny. Například: `#define X_ENABLE_PIN 38`. Touto definicí se říká, že pin 38 bude zastupován názvem `X_ENABLE_PIN`.

V části setup je nastavena funkce jednotlivých pinů. Například: `pinMode(X_ENABLE_PIN, OUTPUT)`. Poté je možné tomuto pinu, který je nastaven jako výstup, přiřadit potřebnou hodnotu. Pin `X_ENABLE_PIN`, jak už jeho název napovídá, povoluje používání příslušného driveru pro ovládání krokového motoru. Pro povolení je potřeba funkcí `digitalWrite(X_ENABLE_PIN, LOW)` zapsat výstupní hodnotu `LOW`.

Následně je nastaven počet disků a věž, na které jsou disky umístěny. Stav má být takový, že na jedné věži jsou všechny disky a na dvou zbývajících není ani jeden.

Nakonec je v části setup proveden inicializační program. Ten pomocí cyklu `while` zajistí najetí levým vozíkem na příslušný koncový spínač, následně zajistí stejnou operaci i pro pravý vozík a cyklus `for` dojede s vozíky do výchozí polohy. V rámci této činnosti jsou nastaveny i proměnné udávající pozice vozíku. Tyto proměnné se jmenují `zleva` a `zprava` a udávají vzdálenost vozíků od koncových spínačů. Vzdálenost je udaná v krocích.

Tyto proměnné s názvy zleva a zprava udávají vzdálenost levého a pravého vozíku od jejich koncových spínačů. Vzdálenost je udávána v krocích motoru.

V části loop jsou cyklicky volány jen tři funkce, z nichž dvě jsou napsány speciálně pro ovládání manipulátoru. Proto nemá smysl se touto částí zabývat před vysvětlením samostatných funkcí.

Funkce jsou brány jako návazné. Tedy na vrcholu stojí hlavní funkce, která řeší samotnou úlohu Hanojských věží. Tato funkce následně volá další funkce pro přesun disků a ty mohou volat další funkce, jako je najetí podavače do určité polohy.

Základním stavebním kamenem tohoto systému jsou čtyři funkce `x_krok_doleva`, `x_krok_doprava`, `y_krok_doleva` a `y_krok_doprava`, kde `x` je levý vozík a `y` pravý vozík. Případně lze pojmout `x` jako horní krokový motor a `y` jako dolní krokový motor. Zavoláním těchto funkcí udělá příslušný krokový motor krok daným směrem (viz Obr. 18).

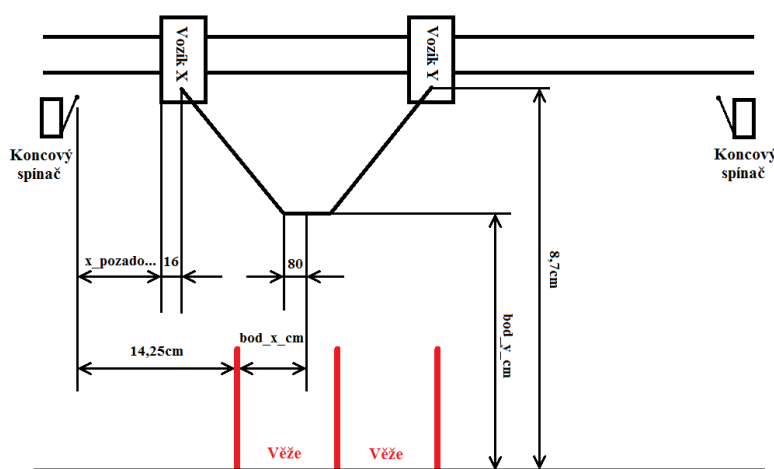
```
//Krokovani leveho jezdcce
void x_krok_doprava(){
    digitalWrite(X_DIR_PIN    , LOW);
    delayMicroseconds(rychlost);
    digitalWrite(X_STEP_PIN   , HIGH);
    delayMicroseconds(rychlost);
    digitalWrite(X_STEP_PIN   , LOW);
}
```

Obr. 18 funkce pro krok levého jezdce směrem doprava

Následuje funkce `pozice`, která zodpovídá za polohování jezdců, a tím pádem i podavače. Vstupními hodnotami této funkce jsou souřadnice bodu v rovinné kartézské soustavě souřadnic zadávané v centimetrech. Zadávají se tedy souřadnice na ose `x` a `y`. Uvedený bod reprezentuje střed podavače disků v ose `x` a spodní plochu magnetu v ose `y`. Počátek souřadného systému je na ose `x` umístěn v ose první věže, na ose `y` pak v patě první věže. Hodnoty na ose `y` rostou, jak se daný bod blíží k vrcholu věže. Na ose `x` se hodnoty zvyšují při pohybu bodu směrem k dalším věžím.

Ve funkci jsou dvě hlavní proměnné $x_pozadovana$ a $y_pozadovana$, které na základě rovnice 2 a do ní vstupujících proměnných bod_y_cm s bod_y_cm udávají vzdálenost vozíků od koncových spínačů. Níže je uvedena rovnice 2 a obr 19, která graficky popisuje hodnoty uvedené v rovnici 2. Všechny použité hodnoty jsou přepočítávány na kroky krokového motoru a výsledná hodnota $x_pozadovana$ udává vzdálenost vozíku x (levý vozík) od jeho koncového spínače v krocích.

$$x_poz=(((14.25+bod_x_cm)*10*4)-(((\cos(\sin((8.7-bod_y_cm-0.3-0.4)/11.85))*11.85)*10*4))-80-16) \quad (2)$$

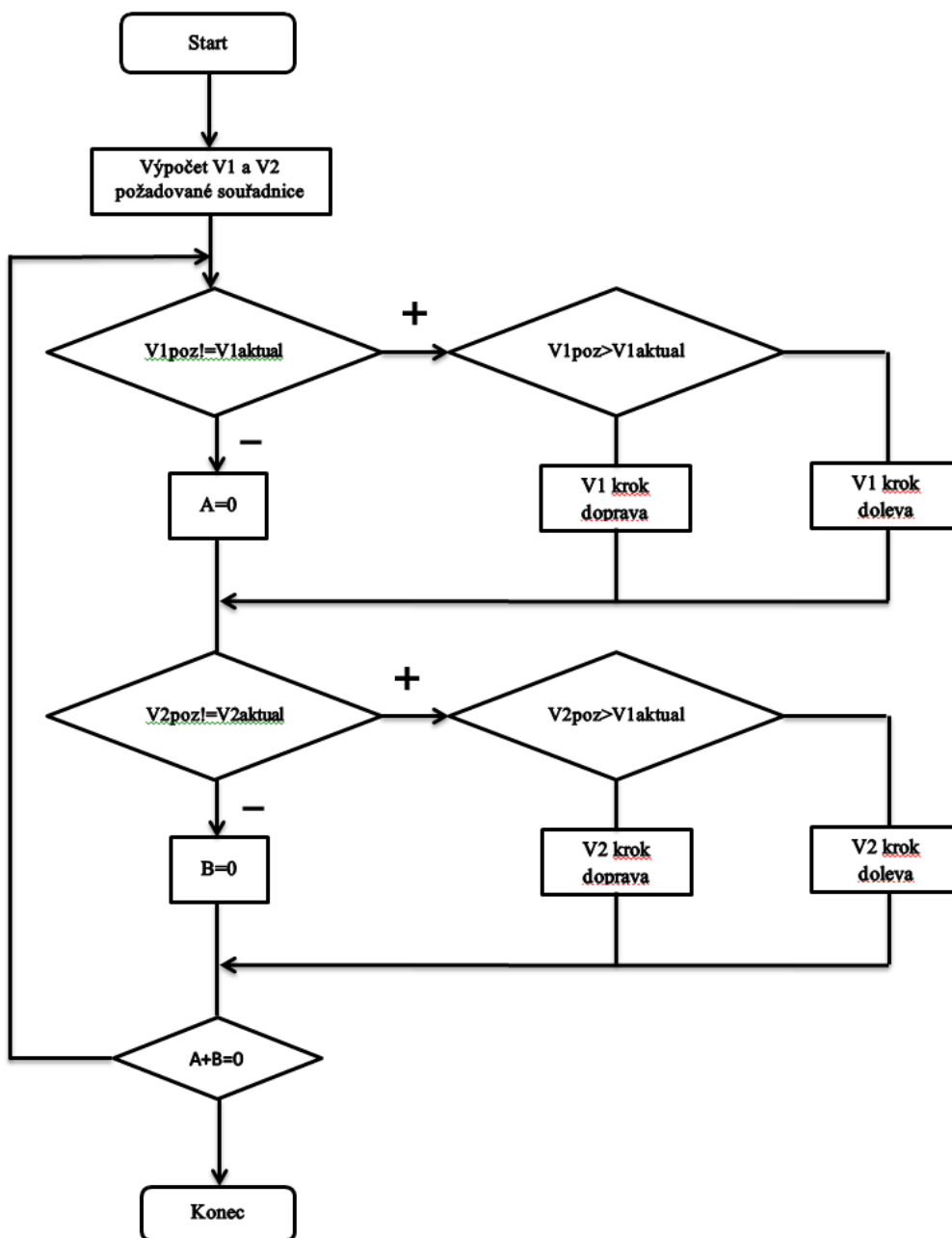


Obr. 19 Vysvětlení konstant a proměnných z rovnice 2. Neoznačené hodnoty jsou v krocích krokového motoru.

Druhou částí funkce pozice je cyklus while s několika podmínkami, který zajistí najetí vozíků do polohy $x_pozadovana$ a $y_pozadovana$.

Na obr. 20 je uveden vývojový diagram pro funkci pozice. Po spuštění jsou v prvním bloku vypočítány dle rovnice 2 proměnné $x_pozadovana$ a $y_pozadovana$. Následně dojde k vnoření do cyklu while, kde je v každém kroku pomocí podmínek „ $V1_poz!=V1_aktual$ “ a „ $V2_poz!=V2_aktual$ “ provedena kontrola, jestli oba vozíky najely do vypočítané polohy. Pokud jsou podmínky splněny a vozíky tedy nejsou v požadované poloze, dojde na uplatnění další podmínky. Pro levý vozík se jedná o podmínku $V1_aktual>V1_poz$. Pokud je tato podmínka splněna, znamená to, že vozík je vpravo od požadované hodnoty a je třeba vykonat krok směrem doleva. Pokud podmínka splněná není, vozík je vlevo od požadované hodnoty a je třeba udělat krok doprava. Podmínka $V2_aktual>V2_poz$ dělá totéž co podmínka $V1_aktual>V1_poz$, jen to dělá pro pravý vozík.

Po najetí vozíku na žádanou pozici dojde k nastavení pomocné proměnné. Pro levý vozík je to proměnná A, pro pravý vozík je to proměnná B. Jakmile budou nastaveny obě proměnné, dojde k ukončení cyklu while a k ukončení celé funkce pozice.



Obr. 20 Vývojový diagram pro funkci pozice

Následuje 6 funkcí, které obstarávají přesun disků mezi jednotlivými věžemi. Název funkce udává, z které a na kterou věž se bude disk přesouvat. Například funkce `presun_2_3()`, přesune disk z věže dva na věž číslo tři. K přesunu používá funkci `pozice`.

Na samém vrcholu tohoto řetězce funkcí stojí funkce `hanoj()`, která má na starosti samotné řešení úlohy Hanojských věží pomocí rekurze. Což znamená, že funkce volá sama sebe.

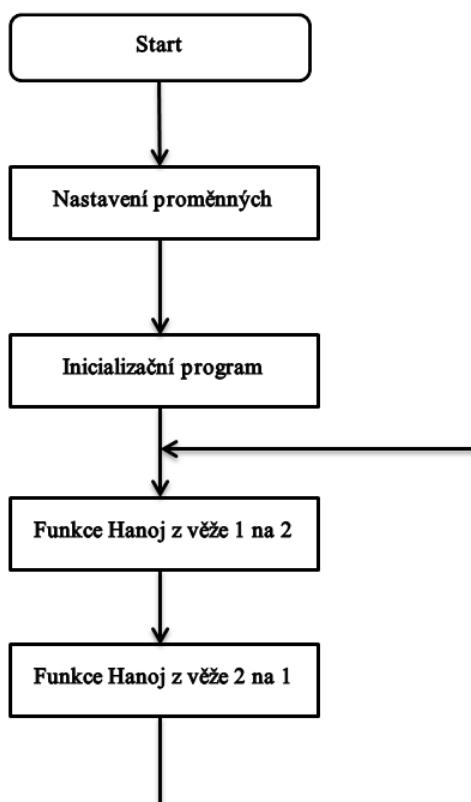
Algoritmus řešení této funkce a zdrojový kód jsou uvedeny v kapitole 3.2 na straně 18.

Poté, co funkce určí, z které věže bude disk zvednut a na kterou věž se disk položí, zavolá funkci `přesun_x_y()` a ta vykoná přesunutí disku. Funkce má 4 vstupní hodnoty. První je počet disků, pak následuje označení věže, na které jsou disky umístěny, následuje označení věže, kam mají být disky přesunuty a nakonec je označení věže, která se používá jako odkládací. Pro logiku funkce je potřeba věže označovat číslicemi od jedné do tří, přičemž první je věž zcela vlevo.

Nyní je možné si vysvětlit obsah smyčky `loop`. V té je nejdříve volána funkce `hanoj(x,1,2,3)`, která přesune x disků z první věže na druhou. Poté se zavolá funkce `pozice(7.1,4)`, která zaparkuje podavač k odkládací věži, čímž se dokončí umístění posledního disku. Následně je možné opět zavolat funkci `hanoj`, tentokrát se však volá se vstupními hodnotami `hanoj(x,2,1,3)`, což způsobí vrácení disků na věž jedna. Po zavolání této funkce se opět volá funkce `pozice`.

Protože je funkce `loop`, v které se volají výše zmíněné funkce, funkcí cyklickou, bude manipulátor neustále přeskládat disky mezi věží jedna a dva.

Aby případný divák postřehl ukončení jedné funkce `hanoj` a začátek druhé, je po každém volání funkce `pozice` zavolána i funkce `delay(5000)`. Tato funkce způsobí zastavení programu na 5 sekund. Vývojový diagram celého programu je na obr. 21.



Obr. 21 Vývojový diagram celého programu

7 ZÁVĚR

Záměrem této bakalářské práce je skrze návrh robotického manipulátoru pro řešení hlavolamu Hanojských věží porozumět základu problematiky robotických manipulátorů. Do návrhu byla zahrnuta volba konceptu manipulátoru. Dále pak jeho realizace a zprovoznění na úrovni, která je vhodná pro reprezentační účely, jako je den otevřených dveří na Fakultě strojního inženýrství VUT Brno.

Prvním drobným krokem bylo pochopení pravidel hlavolamu Hanojských věží a jeho řešení pomocí rekurze. Následující část je již zaměřena na návrh fyzické podoby manipulátoru. Začíná seznámením s již realizovanými manipulátory pro řešení Hanojských věží, které bylo možné nalézt prostřednictvím internetu. Po prozkoumání realizovaných manipulátorů byl navržen vhodný typ konceptu konstrukce. Samotný návrh konstrukce probíhal v 3D modelačním programu, kde bylo možné pracovat s jednotlivými částmi umístěnými v celé sestavě manipulátoru. To umožnilo posuzovat vzájemné ovlivňování součástí a podle potřeby je upravovat. Při návrhu se počítalo, že unikátní části manipulátoru budou vytištěny na 3D tiskárně Rebel II. Po návrhu manipulátoru přišla na řadu volba vhodné elektroniky. Zde byla možnost pro bližší seznámení se světem kolem vývojových kitů Arduino. Pro účely této bakalářské práce byla zvolena kombinace Arduino Mega, RAMPS 1.4 a Pololu DRV8825.

V konečné fázi došlo na fyzickou kompletaci manipulátoru. Manipulátor se kompletoval z nakoupených a vytištěných částí. Celá kompletace manipulátoru se obešla bez výraznějších komplikací. Po ní bylo možné naprogramovat řídicí software pro řešení hlavolamu Hanojských věží prostřednictvím navrženého robotického manipulátoru. Postupovalo se krok po kroku. Nejdříve byly napsány funkce pro ovládání motoru následované funkcí pro polohování podavače. V poslední části programování se přistoupilo k napsání funkce pro řešení hlavolamu Hanojských věží.

Manipulátor nyní úspěšně zvládá řešit hlavolam Hanojských věží o sedmi discích. Řešení provádí cyklicky přeskládáváním disků mezi věží jedna a dvě.

Manipulátor lze dále vylepšit přidáním zobrazovače času a voliče množství disků. Je zde i možnost přidat více disků.

SEZNAM POUŽITÉ LITERATURY

- [1] Édouard Lucas. *Britannica* [online]. 2015 [cit. 2015-05-23]. Dostupné z: <http://www.britannica.com/EBchecked/topic/350482/Edouard-Lucas>
- [2] Hanojské věže. *Wikipedia otevřená encyklopedie* [online]. 2009 [cit. 2015-05-23]. Dostupné z: http://cs.wikipedia.org/wiki/Hanojsk%C3%A9_v%C4%9B%C5%BEE
- [3] NÚÑEZ, André. Arduino Robotic Arm with Inverse Kinematics: Towers of Hanoi. *Youtube* [online]. 27.12.2011 [cit. 2015-05-23]. Dostupné z: http://cs.wikipedia.org/wiki/Hanojsk%C3%A9_v%C4%9B%C5%BEE
- [4] ERCOST60,. \$50 Robot Arm Solves Tower of Hanoi. *Youtube* [online]. 25.9.2010 [cit. 2015-05-23]. Dostupné z: <https://www.youtube.com/watch?v=-Z8ITSX4PHs>
- [5] VAN DE VEEN, Paul. Towers of Hanoi robot Meccano. *Youtube* [online]. 24.7.2014 [cit. 2015-05-23]. Dostupné z: <https://www.youtube.com/watch?v=dxWFjYhuLr4>
- [6] EDWARD, Jim. Tower of Hanoi. *Youtube* [online]. 26.1.2008 [cit. 2015-05-23]. Dostupné z: <https://www.youtube.com/watch?v=RPuzuolav2s>
- [7] MAKEBLOCK,. Makeblock: Tower of Hanoi [demo]. *Youtube* [online]. 23.9.2013 [cit. 2015-05-23]. Dostupné z: <https://www.youtube.com/watch?v=Btw3t9fxvuQ>
- [8] MALÝ, Martin. Arduino: vývojový kit pro hrátky s hardware. *Root.cz* [online]. 2009 [cit. 2015-05-23]. Dostupné z: <http://www.root.cz/clanky/arduino-vyvojovy-kit-pro-hratky-s-hardware/>
- [9] Arduino Mega 2560. *Arduino* [online]. 2015 [cit. 2015-05-23]. Dostupné z: <http://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- [10] RAMPS 1.4. *RepRap* [online]. 2011 [cit. 2015-05-23]. Dostupné z: http://reprap.org/wiki/RAMPS_1.4#Wiring
- [11] DRV8825 Stepper Motor Driver Carrier, High Current (md20a). *Pololu Robotic & Electronics* [online]. 2015 [cit. 2015-05-23]. Dostupné z: <https://www.pololu.com/product/2132>
- [12] CZ.ARDUINO,. Lekce 1: Začínáme s Arduinem. *Arduino8* [online]. 2012 [cit. 2015-05-23]. Dostupné z: <http://arduino8.webnode.cz/news/zaciname-s-arduinem/>

SEZNAM PŘÍLOH:

Příloha č.1 CD s tímto obsahem:

- Bakalářská práce na téma Robotický manipulátor pro řešení hanojských věží
- Zdrojový kód řídicího programu v arduino file a textové podobě