

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

ŘÍZENÍ CNC FRÉZKY

CNC MILL CONTROL SYSTEM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Luděk Hájek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jakub Arm

BRNO 2018



Bakalářská práce

bakalářský studijní obor **Automatizační a měřicí technika**
Ústav automatizace a měřicí techniky

Student: Luděk Hájek

Ročník: 3

ID: 173650

Akademický rok: 2017/18

NÁZEV TÉMATU:

Řízení CNC frézky

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je vytvoření řízení CNC frézky pomocí LinuxCNC.

1. Provedte návrh komponent pro CNC frézku.
2. Provedte rešerši řídicích systémů pro hobby CNC frézky.
3. Navrhněte a realizujte řídicí systém CNC frézky.
4. Demonstrujte funkčnost řešení a celého procesu od návrhu výrobku až po jeho realizaci.
5. Zdokumentujte nastavení LinuxCNC a konfiguraci celého systému.

DOPORUČENÁ LITERATURA:

Literatura projektu LinuxCNC.

Termín zadání: 5. 2. 2018

Termín odevzdání: 21.5.2018

Vedoucí práce: Ing. Jakub Arm

Konzultant:

doc. Ing. Václav Jirsík, CSc.
předseda oborové rady



UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Abstrakt

Tato bakalářská práce se zabývá návrhem konstrukce a řízení 3-osé hobby CNC frézky pomocí LinuxCNC. Všechny osy jsou poháněny krokovými motory Microcon SX 24-2341, které jsou napájeny drivery Leadshine M542. LinuxCNC je nainstalován na PC s real-time verzí Ubuntu 12.04.5 LTS a s drivery je propojen pomocí breakout boardu. Pro grafický návrh výrobku, vytváření cest nástroje a generování G-kódu je zvolený CAD/CAM program Autodesk Fusion 360. Součástí práce je rešerše v oblasti konstrukčních prvků a řízení CNC strojů, výpočet a výběr vhodných pohonů, návrh a následné zdokumentování zvoleného řídicího systému frézky, včetně zdokumentování možností a nastavení programů LinuxCNC a CAM části programu Fusion 360 spolu s příkladem hotového výrobku.

Klíčová slova

Lineární vedení, krokový motor, CNC kontrolér, breakout board, driver, interpolátor, CAD, CAM, LinuxCNC

Abstract

This Bachelor thesis deals with designing construction and control system of 3-axis hobby CNC router using LinuxCNC. All axes are driven by Microcon SX 24-2341 stepper motors, which are provided with power from Leadshine M542 stepper drivers. LinuxCNC is installed on real-time PC with Ubuntu 12.04.5 LTS and it is connected with drivers via breakout board. CAD/CAM software Autodesk Fusion 360 is used for graphical designing of final product, creating toolpaths and generating G-code. The Bachalors thesis contains research of construction elements, possibilities of CNC controlling, calculation and selection of suitable motors, design and documentation of CNC control system including documentation of all settings of LinuxCNC and CAM part of Fusion 360 as well as example of final product, created using this CNC router.

Keywords

Linear slide, stepper motor, CNC controler, breakout board, driver, interpolator, CAD, CAM, LinuxCNC

Bibliografická citace:

HÁJEK, L. Řízení CNC frézky. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2018. 69 s. Vedoucí bakalářské práce Ing. Jakub Arm.

Prohlášení

Prohlašuji, že svou bakalářskou práci na téma Řízení CNC frézky jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: 20. května 2018

.....
podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Jakubu Armovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Zároveň bych rád poděkoval mému otci Lud'ku Hájkovi za pomoc s mechanickou konstrukcí CNC frézky a poskytnutí celkového technického zázemí.

V Brně dne: 20. května 2018

.....
podpis autora

Obsah

Úvod	10
1 Mechanická konstrukce.....	11
1.1 Základní požadavky na CNC frézku	11
1.2 CNC frézka s pohyblivým portálem	11
1.3 CNC frézka s pevným portálem	12
1.4 Lineární vedení.....	13
1.4.1 Nepodepřené tyče.....	13
1.4.2 Podepřené tyče	13
1.4.3 Prizmatické vedení.....	14
1.5 Pohybový mechanismus lineárního posunu	15
1.5.1 Šrouby s lichoběžníkovým závitem (Trapézový šroub)	15
1.5.2 Kuličkové šrouby.....	15
1.5.3 Ozubené řemeny a hřebeny	16
2 Motory.....	17
2.1 Servopohon.....	17
2.2 Stejnoseměrné motory.....	17
2.2.1 Elektronicky komutované motory	18
2.3 Asynchronní motory.....	18
2.4 Synchronní motory	18
2.5 Krokové motory (KM).....	19
2.5.1 Způsoby řízení krokových motorů.....	20
2.6 Výpočet a výběr vhodného motoru	23
3 Řídicí elektronika.....	27
3.1 Počítač.....	27
3.2 CNC kontrolér.....	28
3.2.1 Breakout board	29
3.2.2 Drivery krokových motorů.....	29
3.3 Další HW komponenty	32
3.3.1 Interpolátor	32
3.3.2 Smoothieboard.....	33
4 Software	34
4.1.1 CAD software	34
4.1.2 CAM software	34
4.1.3 Interpret G-kódu – ovládací software pro CNC frézku	35
5 Použité zapojení.....	37
5.1 Řídicí elektronika.....	37
5.2 PC.....	38

5.3	LinuxCNC.....	39
5.3.1	Latency test + výběr finálního PC	39
5.3.2	Nastavení parametrů.....	40
5.3.3	Uživatelské prostředí – axis GUI	45
5.4	CAD/CAM	47
5.4.1	Google SketchUp + SketchUcam.....	48
5.4.2	Autodesk Fusion 360	49
6	Zkušební frézování a plány do budoucna.....	58
	Závěr.....	60
	Literatura	62
	Seznam symbolů, veličin a zkratk.....	64
	Seznam příloh.....	65

Seznam obrázků

<i>Obrázek 1.1: CNC frézka s pohyblivým portálem [2]</i>	12
<i>Obrázek 1.2: CNC s pevným portálem [2]</i>	12
<i>Obrázek 1.3: Nepodepřená tyč a jednoduchý uzavřený linearset [3]</i>	13
<i>Obrázek 1.4: Podepřená tyč a jednoduchý otevřený linearset [3]</i>	14
<i>Obrázek 1.5: Prizmatické vedení s vozíkem [3]</i>	14
<i>Obrázek 1.6: Trapézový šroub a bronzová matice [4]</i>	15
<i>Obrázek 1.7: Kuličkový šroub a matice [4]</i>	16
<i>Obrázek 1.8: ozubený hřeben (vlevo) [5] a ozubený řemen (vpravo) [6]</i>	16
<i>Obrázek 2.1: Řez stejnosměrným motorem s permanentními magnety [7]</i>	17
<i>Obrázek 2.2: Řez a princip funkce elektronicky komutovaného motoru [8]</i>	18
<i>Obrázek 2.3: Konstrukce reluktančního krokového motoru[11]</i>	19
<i>Obrázek 2.4: Konstrukce krokového motoru s permanentním magnetem[11]</i>	20
<i>Obrázek 2.5: Konstrukce hybridního krokového motoru[11]</i>	20
<i>Obrázek 2.6: Bipolární a unipolární zapojení dvoufázového hybridního krokového motoru [10]</i>	21
<i>Obrázek 2.7: Momentová charakteristika motoru SX24-2341 s CD30x, CD30M (48 VDC), SX24-3341 s CD40x, CD40M (48VDC)[13]</i>	22
<i>Obrázek 3.1: Schéma řízení CNC frézky, kde je každý driver KM napájen samostatně ze zdroje [14]</i>	29
<i>Obrázek 3.2: Schéma řízení CNC frézky, kde jsou drivery napájeny pomocí breakout boardu, který slouží jako rozvod napájení ze zdroje [14]</i>	29
<i>Obrázek 5.1 Výsledek Latency testu pro konečnou sestavu</i>	40
<i>Obrázek 5.2: LinuxCNC - Base configuration</i>	41
<i>Obrázek 5.3: LinuxCNC - nastavení Paralelního portu</i>	42
<i>Obrázek 5.4: LinuxCNC - nastavení parametrů pro jednotlivé osy</i>	44
<i>Obrázek 5.5: LinuxCNC - axis GUI [22]</i>	46
<i>Obrázek 5.6: Fusion 360 - CAD</i>	49
<i>Obrázek 5.7: Fusion 360 - CAM</i>	50
<i>Obrázek 5.8: Fusion 360 - CAM - New setup</i>	51
<i>Obrázek 5.9: Fusion 360 - nastavení parametrů 2D Contour</i>	53
<i>Obrázek 5.10: Fusion 360 - Simulace</i>	55
<i>Obrázek 5.11: Fusion 360 - nastavení Post processingu</i>	56
<i>Obrázek 5.12: Editace vygenerovaného G-kódu</i>	57

Seznam tabulek

Tabulka 2.1: Změřené rozměry a hmotnosti jednotlivých součástí CNC frézky 23

ÚVOD

Obsahem mé bakalářské práce je návrh konstrukce a řízení tříosé hobby CNC frézky. Tuto frézku jsem zkonstruoval doma na dílně a sloužit bude hlavně na frézování dřeva. Stroj má rozměry pojezdů 1000x600x300 mm a je poháněn krokovými motory Microcon SX 24-2341. K řízení frézky je použit program LinuxCNC, který je nainstalován na stolním PC. PC generuje STEP/DIR pulzy a přes breakout board dodává signály driverům pro krokové motory Leadshine M542. K vytváření grafických návrhů a následně G-kódu je použit CAD/CAM program Autodesk Fusion 360.

Bakalářská práce je rozdělena do šesti hlavních kapitol.

V první kapitole jsem provedl rešerši mechanické konstrukce stroje, vedení jednotlivých os a možných pohybových mechanismů pro přenos rotačního pohybu motorů na pohyb lineární.

Druhá kapitola obsahuje rozbor typů motorů používaných při stavbách CNC frézek, jejich výhody a nevýhody a následně výpočet krouticího momentu motoru, potřebného pro mou frézku spolu s výběrem konkrétního vhodného pohonu.

Ve třetí kapitole jsem provedl rešerši řídicích systémů frézky. Jakým způsobem se CNC frézka může řídit, jaké všechny komponenty jsou k tomu zapotřebí, jejich hlavní parametry, výhody a nevýhody.

Ve čtvrté kapitole se věnuji rešerši potřebného řídicího softwaru, jaké skupiny programů se používají při vytváření a výrobě výrobku na CNC frézce a nepoužívanější programy z těchto skupin.

Pátá kapitola obsahuje kompletní popis finálního zapojení na základě získaných informací. Začínám popisem použité řídicí elektroniky a jejich parametrů spolu s popisem PC, na kterém běží SW LinuxCNC. Následuje detailní popis nastavení samotného programu LinuxCNC spolu s výčtem některých problémů a řešení, na které jsem narazil při stavbě a zapojování stroje. Kapitulu uzavírá odůvodnění výběru a popis zvoleného CAD/CAM SW, včetně nastavení CAM. Práce je tedy seřazena chronologicky od nejnižší HW vrstvy po nejvyšší SW vrstvu tak, aby bylo možné jednoduše zpětně zjistit, jak postup výroby funguje od obrázku po výrobek.

V šesté, závěrečné kapitole se věnuji popisu některých dalších problémů, co vznikly při konstrukci a plánů do budoucna na úpravy a vylepšení stroje.

1 MECHANICKÁ KONSTRUKCE

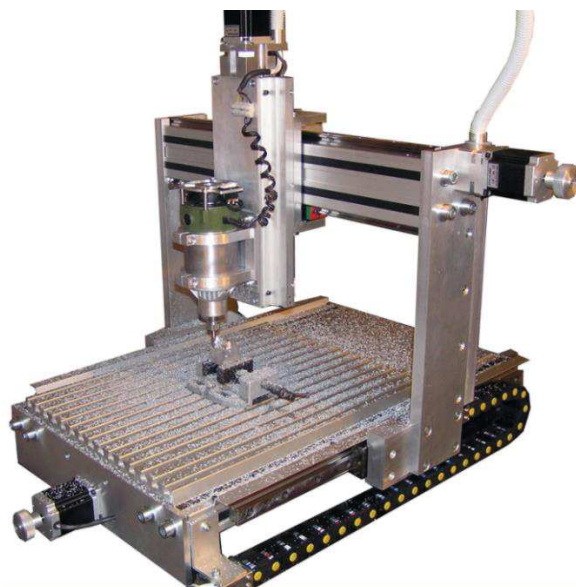
Prvním bodem mé práce byla volba mechanické konstrukce CNC frézky. Již na začátku jsem věděl, že budu stavět CNC frézku portálového typu (2 stojany/ bočnice spojené příčnickem = portál). Na výběr jsem měl ze dvou možností (S pohyblivým nebo s pevným portálem). Oba typy frézky operují ve třech osách, což bylo mým základním parametrem.

1.1 Základní požadavky na CNC frézku

Mým zámyslem bylo vyrobit stroj určený na opracovávání dřeva, plastu a po odzkoušení případně i hliníku. Odzkoušení je zde nutné proto, že konstrukce frézky je vyrobena z materiálů dostupných na dílně a jestli bude dostatečně tuhá a robustní se ukáže až v praxi. Primárně tedy je stroj zaměřen na opracovávání dřevěných a plastových obrobků s malým úběrem a nízkými rychlostmi posuvu s důrazem na cenu jednotlivých komponent. Jedná se o hobby CNC frézku, kterou jsem zkonstruoval doma na dílně. Požadavek byl, aby se jednalo o středně velký stroj, na kterém by bylo možné opracovávat i větší a těžší bloky dřeva, případně jiných materiálů. Zvolil jsem proto konstrukci s pohyblivým portálem a rozměry pojezdů 1000 x 600 x 300 mm.

1.2 CNC frézka s pohyblivým portálem

Pohyb ve všech třech osách zajišťuje pohyblivý portál. Stůl frézky, na kterém se nachází obrobek, je pevně spojen s konstrukcí a nepohybuje se. Díky tomu, že pohybuje celým portálem, jsou na něj kladeny velké nároky na tuhost. Pokud by portál nebyl dostatečně tuhý, mohlo by docházet k rezonancím a trhanému pohybu, což by mělo vliv na přesnost a použitelnost celé frézky. Hlavní výhodou tohoto konstrukčního řešení je možnost práce s velkými a těžkými obrobky, jelikož zde váha obrobku nehraje roli. Ve své stavbě jsem se rozhodl použít tuhle konstrukci právě z tohoto důvodu.



Obrázek 1.1: CNC frézka s pohyblivým portálem [2]

1.3 CNC frézka s pevným portálem

Jedná se o typ frézky, kdy je portál pevně spojen s konstrukcí stolu a vřeteno na něm upnuté se pohybuje pouze výškově a do stran. Stůl, na kterém se nachází obrobek, se rovněž pohybuje v jedné ose. Tento typ frézky umožňuje nižší nároky na tuhost celé konstrukce. Je vhodný pro obrábění menších obrobků s nízkou hmotností, jelikož stůl se pohybuje spolu s obrobkem. Hodí se tak např. pro výrobu DPS, modelářských dílů, atd..



Obrázek 1.2: CNC s pevným portálem [2]

1.4 Lineární vedení

V rámci mé konstrukce CNC frézky jsem se rozhodoval mezi třemi základními typy lineárního vedení. Mezi základní parametry, které mi pomohly při výběru správného typu, patří hlavně dostatečná tuhost vedení pro danou délku pojezdu a váhu, kterou vedení ponese a cena jednotlivých variant.

1.4.1 Nepodepřené tyče

Nepodepřené tyče jsou nejlevnější variantou lineárního vedení. Jedná se o broušenou, povrchově kalenou tyč, po které se pohybují posuvná pouzdra (uzavřené vozíky = Linearset). Tato pouzdra mají vevnitř kuličky, které slouží jako ložiska a vozík se po nich pohybuje po tyči. Hlavní výhodou jejich použití je nízká pořizovací cena. Hodí se hlavně na místa, kde nejsou velké vzdálenosti přejezdů a nejsou kladeny velké nároky na tuhost pojezdu. Při použití na delší vzdálenosti by docházelo k průhybu tyčí, a proto je tento typ vedení pro tyto aplikace nepoužitelný.



Obrázek 1.3: Nepodepřená tyč a jednoduchý uzavřený linearset [3]

1.4.2 Podepřené tyče

Podepřené tyče představují ideální kompromis mezi tuhostí a cenou. Jedná se o nejpoužívanější typ lineárního vedení při konstrukci větších hobby CNC strojů. Vedení je koncipováno podobně jako u nepodepřených tyčí, avšak tyč je vyztužena podepřením, což několikanásobně zvyšuje její tuhost. Po tyči se pohybují otevřené vozíky pracující na stejném principu jako u nepodepřených tyčí. Tento typ vedení je možná použít i na velké vzdálenosti a větší zatížení, jelikož díky ztužení nedochází k průhybu tyčí. Rovněž cena je stále přijatelná. Pro svou stavbu jsem zvolil právě tento typ vedení, díky dostatečné tuhosti i pro rozměry mého stroje a rozumné pořizovací ceně.



Obrázek 1.4: Podepřená tyč a jednoduchý otevřený lineáreset [3]

1.4.3 Prizmatické vedení

Jedná se o nejdražší a nejpřesnější typ lineárního vedení. Hlavní výhodou je vysoká přesnost vedení, nízký součinitel tření a vysoká povolená zátěž na vozík při jeho menších rozměrech. Používá se ve strojích, kde je zapotřebí vysoké přesnosti a plynulosti lineárního pohybu, nebo při požadavku malých rozměrů vedení (3D tiskárny, ...). Vysoká kvalita a přesnost se ovšem negativně promítne na několikanásobně vyšší ceně těchto komponent v porovnání s výše zmíněnými možnostmi.



Obrázek 1.5: Prizmatické vedení s vozíkem [3]

1.5 Pohybový mechanismus lineárního posunu

Pod pojmem pohybový mechanismus lineárního posunu mám na mysli takový mechanismus, který přeměňuje rotační pohyb motoru na lineární pohyb jednotlivých os. Takovýchto mechanismů existuje hned několik, avšak pro použití u CNC frézky se jich hodí jen pár. Pro použití u CNC frézky je stěžejních několik parametrů pohybových mechanismů a to hlavně co nejmenší a konstantní hodnota součinitele tření, tuhost mechanismu a minimální vůle. Z tohoto důvodu zde uvádím pouze tři hlavní a nejčastěji používané způsoby pohybových mechanismů pro použití v této oblasti.

1.5.1 Šrouby s lichoběžníkovým závitem (Trapézový šroub)

Jedná se o nejlevnější variantu pohybového mechanismu, nejčastěji používanou u levných hobby CNC strojů. Používají se tam, kde nejsou kladeny velké nároky na přesnost polohování, a není potřeba velkých posuvových rychlostí. Tento mechanismus se vyznačuje nízkou účinností, vůlí v závitech a vysokým opotřebením šroubu. Matice jsou nejčastěji z plastu nebo bronzu.



Obrázek 1.6: Trapézový šroub a bronzová matice [4]

1.5.2 Kuličkové šrouby

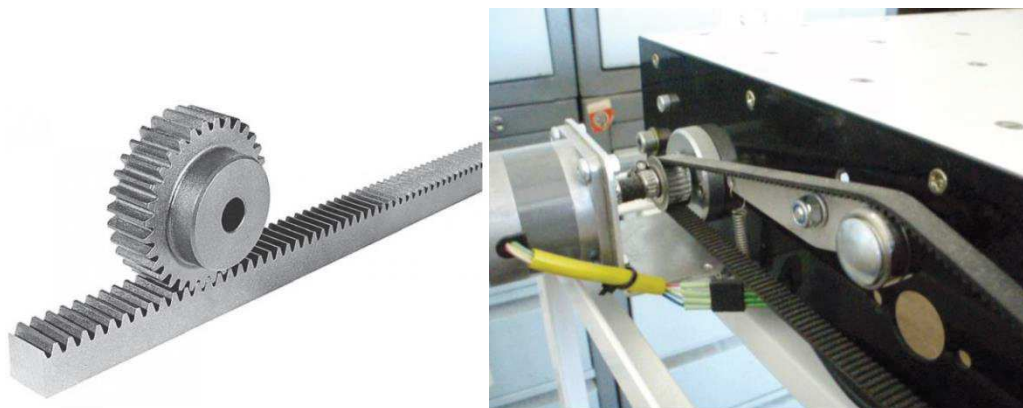
V současné době nejpoužívanější varianta pohybového mechanismu. V závitech mezi maticí a šroubem se nacházejí kuličky, díky kterým se výrazně eliminuje koeficient tření a třecí odpor se převádí na valivý. Matice je uzpůsobena tak, aby v ní kuličky neustále obíhaly a vracely se do oběhu díky drážkám v jejím plášti. Tento šroub se vyznačuje vysokou účinností (až 93%), vysokou tuhostí a přesností a díky předepnutí matice téměř nulovou vůlí. Nevýhodou je vyšší cena díky složitější výrobě. Díky stále klesajícím cenám kuličkových šroubů a velikosti konstrukce mé CNC frézky jsem zvolil tento typ pohybového mechanismu.



Obrázek 1.7: Kuličkový šroub a matice [4]

1.5.3 Ozubené řemeny a hřebeny

Oba typy pohybového mechanismu se používají v krajních případech, kdy jsou na systém kladeny speciální nároky. Ozubené řemeny používáme tam, kde je potřeba velkých rychlostí přejezdů. Ozubené hřebeny se zase používají v aplikacích, kde je potřeba se pohybovat na velké vzdálenosti. Oba typy ovšem mají v porovnání s klasickými šrouby řádově nižší přesnost polohování.



Obrázek 1.8: ozubený hřeben (vlevo) [5] a ozubený řemen (vpravo) [6]

2 MOTORY

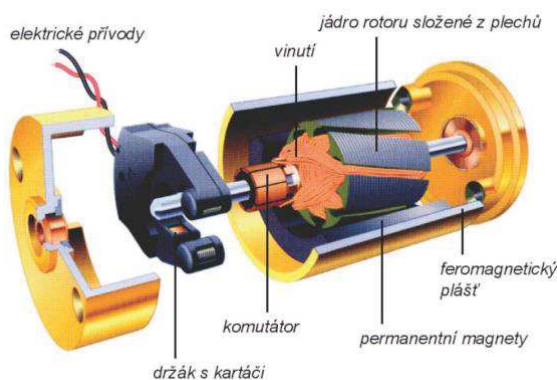
2.1 Servopohon

Pod pojmem elektrický servopohon v mechatronice rozumíme subsystém, který zajišťuje přeměnu elektrické energie v mechanický pohyb a jeho řízení. Elektrický servopohon jako systém obsahuje elektrický motor (motory) a řídicí a napájecí elektroniku se zpětnou vazbou. Podle typu zpětné vazby, kterou z motoru dostáváme, můžeme dělit servopohony na polohové (dostáváme údaj o poloze natočení hřídele) a rychlostní (údaj o rychlosti otáčení hřídele). [16]

Dále zde nastíním základní dělení motorů v servopohonech, princip jejich funkce a jejich výhody a nevýhody.

2.2 Stejnospměrné motory

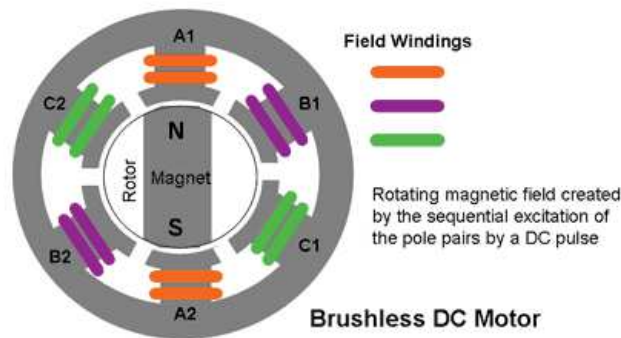
Stejnospměrný motor se skládá ze statoru, rotoru a sběracího ústrojí. Stator je tvořený permanentními magnety a pólovými nástavci z magneticky měkkého železa. Ty koncentrují magnetický tok do vzduchové mezery. Rotor je tvořen jednotlivými cívkami uloženými v drážkách. Tyto cívky jsou vyvedeny na komutátor, který spolu s uhlíkovými kartáčky slouží jako sběrací ústrojí. Tento motor se velmi dobře reguluje (otáčky měníme změnou přiváděného napětí), avšak nevýhodou je jiskření v oblasti komutátoru a opotřebení sběracího ústrojí (hlavně uhlíkových kartáčků) a nutnost jejich následné výměny.



Obrázek 2.1: Řez stejnosměrným motorem s permanentními magnety [7]

2.2.1 Elektronicky komutované motory

Častěji používaným typem stejnosměrného motoru je elektronicky komutovaný stejnosměrný motor (též BLDC (BrushLess DC), či ECM). Jeho stavba je velmi podobná klasickému stejnosměrnému motoru, avšak BLDC má magnety na rotoru a vinutí ve statoru. Díky tomu odpadá nutnost použití sběracího ústrojí a celá jeho náchylnost na poruchy a opotřebení. Řídicí elektronika podle polohy rotoru přepíná proud do jednotlivých vinutí statoru. Poloha rotoru se snímá diskrétně (magneticky – Hallův snímač, nebo fotoelektricky), vždy po 60-ti stupních.



Obrázek 2.2: Řez a princip funkce elektronicky komutovaného motoru [8]

2.3 Asynchronní motory

Stator asynchronního motoru je tvořen vinutím. Rotor je buď klecový (s vinutím = hliníkové nebo měděné tyče na koncích spojené), nebo kroužkový (s vinutím izolovanými vodiči v rotorových drážkách, vyvedenými na rotorové sběrací kroužky). Tyto motory také obsahují snímače otáček nebo polohy. Otáčky lze regulovat změnou počtu pólů (skokově), nebo změnou frekvence napájecího napětí (beze ztrát). Jedná se o velmi rozšířený typ motoru používaný pro pohon velkých pracovních strojů.

2.4 Synchronní motory

Rotor synchronního motoru se otáčí synchronně s otáčením točivého magnetického pole statoru. Stator je listěný, s drážkami s vinutím, které je zpravidla třífázové. V servopohonech se používá typ synchronních motorů, které jsou buzeny magnety na rotoru (buď na povrchu, nebo zapuštěné). Konstrukčně je tento typ podobný BLDC motoru. Hlavní rozdíl mezi nimi je ve snímání polohy rotoru. BLDC motor snímá polohu rotoru pouze diskrétně po 60-ti stupních, zatímco stejnosměrný motor používá resolver, který snímá polohu rotoru spojitě. Hlavní výhodou oproti asynchronním motorům je to, že při zachování výkonu jsou

synchronní motory lehčí, menší a lépe se chladí. Jedná se o nejrozšířenější typ motorů pro automatizaci, výrobní roboty a polohovací techniku.

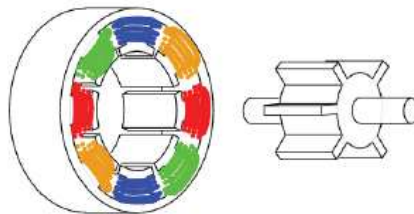
2.5 Krokové motory (KM)

Krokové motory jsou nejrozšířenější motory pro aplikaci u menších, počítačem řízených strojů (hobby CNC, 3D tiskárny, ...). Pohyb KM je diskretní, pohybuje se po jednotlivých krocích. Tyto kroky jsou vyvolávány impulzy do statorových vinutí, které jsou do motoru dodávány řídicí elektronikou. KM se skládají ze statoru a rotoru, který je na obou koncích uložen v ložiscích.

Existují 3 základní typy KM:

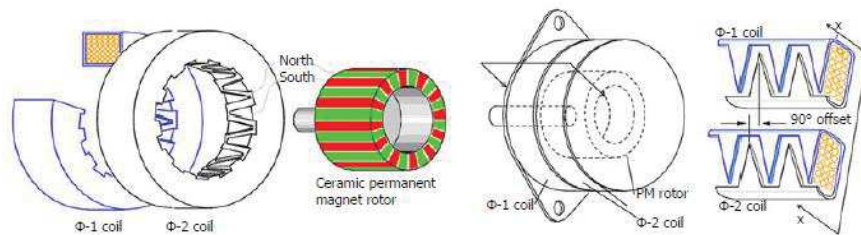
- Reluktanční KM (krokový motor VR)
- KM s permanentním magnetem (krokový motor PM)
- Hybridní KM

Reluktanční KM má stator s nástavci s vinutím a rotor z magneticky měkkého železa, vybavený řadou zubů. Postupně přivádíme napětí na jednotlivé cívky statoru a tím dochází k postupnému přitahování zubů rotoru k jednotlivým cívkám a jeho otáčení. Jeho předností jsou dobré dynamické vlastnosti, díky malému momentu setrvačnosti rotoru a nízká cena. Nevýhodou je ovšem velký krok (nejčastěji 30° na krok). Díky tomu se nehodí pro aplikace, kde je potřeba přesného polohování.



Obrázek 2.3: Konstrukce reluktančního krokového motoru[11]

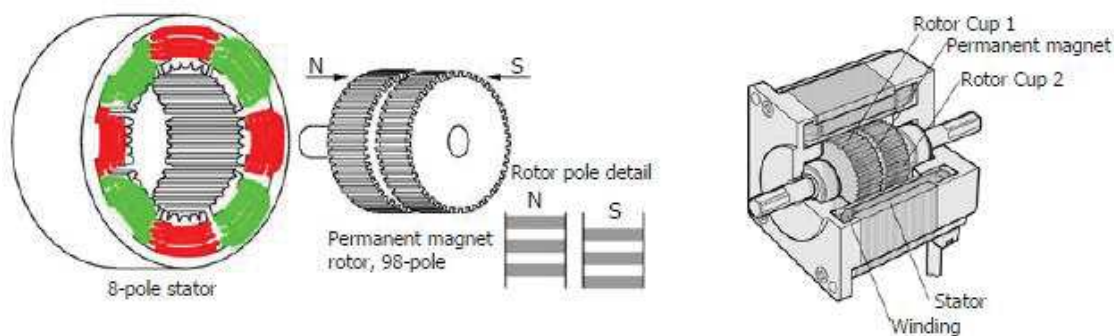
KM s permanentním magnetem má rotor bez zubů, tvořený permanentním magnetem a stator nástavci s vinutím se zuby, které jsou ohnuty směrem k rotoru. Princip funkce je obdobný jako u reluktančního KM, kdy postupně přivádíme napětí na jednotlivé cívky statoru, abychom docílili otáčení. Rozlišení motoru (počet kroků) je zde úměrný počtu pólů magnetu statoru. Typicky se tyto motory vyrábí s krokem 7,5° až 15°.



11

Obrázek 2.4: Konstrukce krokového motoru s permanentním magnetem[11]

Hybridní KM má rotor tvořený permanentním magnetem se zuby. Rotor se dělí na dva segmenty, kdy v prvním jsou severní a v druhém jižní póly magnetu a tyto segmenty jsou vzájemně posunuty o půl zubu. Rotor má běžně 50 zubů. Stator má opět pólové nástavce se zuby a vinutím. Nejčastěji se jedná o 8 pólů ve statoru, kdy kolem každého pólu statoru jsou 2 vinutí. Hybridní motory mají nejčastěji při řízení ve „full step“ módu 200 kroků na otáčku, což znamená úhel pootočení $1,8^\circ$ na krok. Tento úhel se dá ještě několikanásobně zmenšit pomocí mikrokrokování, které rozeberu později. Hybridní motory se v dnešní době používají čím dál více, díky levné výrobě v Číně a stávají se tak dostupné i pro hobby použití. Nejčastěji se používají 2-fázové hybridní KM, ale dají se koupit i 3-fázové, které mají ještě menší úhel pootočení na jeden krok. Hybridní KM jsou obecně nejlepší variantou pro použití v hobby CNC frézkách, díky jejich ceně, velmi malému kroku, velkému momentu a možnosti relativně vysokých otáček.



Obrázek 2.5: Konstrukce hybridního krokového motoru[11]

2.5.1 Způsoby řízení krokových motorů

Volbou způsobu řízení krokových motorů můžeme ovlivnit některé jejich vlastnosti. Zejména se jedná o velikost kroku, rychlost otáčení či krouticí moment motoru. Nyní zde popíšu základní druhy zapojení a řízení krokových motorů a jejich vliv na vlastnosti motoru.

2.5.1.1 Unipolární vs. Bipolární řízení

Unipolární řízení

Při unipolárním řízení, prochází proud vždy právě jednou cívkou motoru. Výhodou takového zapojení je menší spotřeba elektrické energie a jednodušší zapojení. Nevýhodou je ovšem menší krouticí moment motoru. [9]

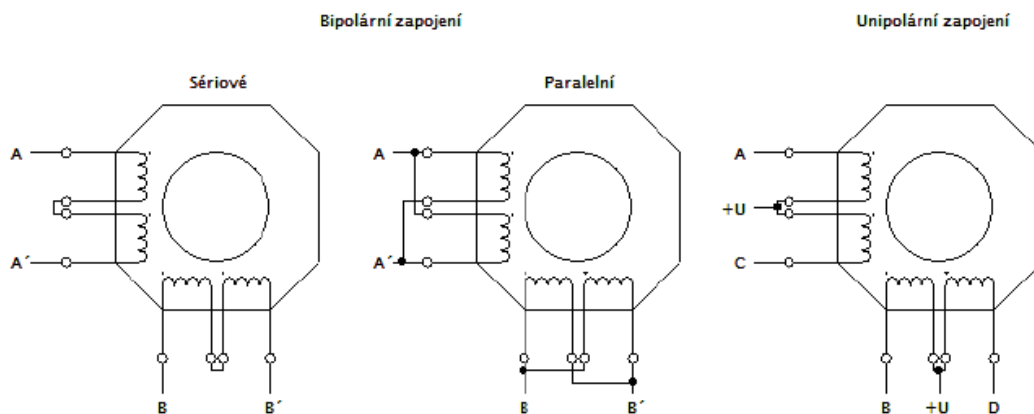
Bipolární řízení

Při bipolárním zapojení prochází proud vždy dvěma protilehlými cívkami a to tak, že každá z protilehlých cívek generuje opačně orientované magnetické pole. Výhodou takového zapojení je větší krouticí moment motoru, nevýhodou ovšem je složitější zapojení řídicí elektroniky. [9]

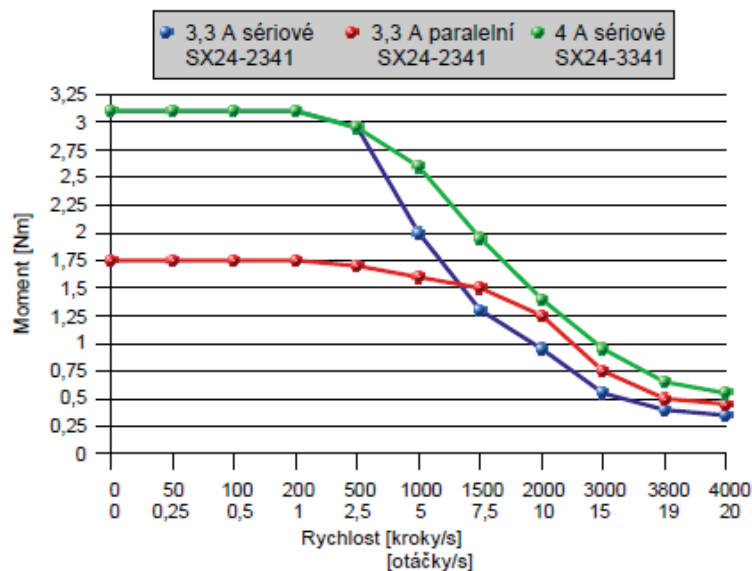
Pokud máme krokový motor, který má vyvedených 8 vodičů, je možné takový motor zapojit několika způsoby. Při bipolárním buzení takového motoru, je možné jej zapojit sériově nebo paralelně.

Při sériovém zapojení dosáhneme toho, že krouticí moment motoru bude v oblasti nižších otáček vyšší a naopak se zvyšujícími se otáčkami bude více klesat.

Paralelní zapojení se hodí naopak pro oblasti vyšších otáček, kde má motor v tomto zapojení vyšší krouticí moment. Nevýhodou ovšem je, že v oblasti nižších krokovacích frekvencí dosáhneme výrazně nižšího krouticího momentu, než při sériovém zapojení.



Obrázek 2.6: Bipolární a unipolární zapojení dvoufázového hybridního krokového motoru [10]



Obrázek 2.7: Momentová charakteristika motoru SX24-2341 s CD30x, CD30M (48 VDC), SX24-3341 s CD40x, CD40M (48VDC)[13]

2.5.1.2 Jednofázové vs. Dvoufázové řízení

Při jednofázovém řízení generuje magnetické pole pouze jedna cívka (při bipolárním řízení dvě cívky). Nejbližší zub rotoru je tedy ke statoru přitahován pouze v jednom místě, což znamená sice menší spotřebu, ale zároveň menší moment motoru. [9]

Při dvoufázovém řízení je magnetické pole shodné orientace generováno vždy dvojicí sousedních cívek. Nejbližší zub rotoru je tedy ke statoru přitahován tak, že jeho klidová poloha je mezi dvěma cívkami. Výhodou je vyšší moment, za cenu vyšší spotřeby. [9]

2.5.1.3 Plný krok, poloviční krok, mikrokrokování

Při řízení s plným krokem dosáhneme takového rozlišení, jaké nám fyzicky povoluje konstrukce motoru (počet zubů rotoru). Motor můžeme řídit jakýmkoliv výše zmíněným způsobem tak, že vždy při jednom kroku otočíme rotorem o jeden celý krok (např. při jednofázovém unipolárním řízení se nejbližší rotorový zub posune o jednu celou polohu.)

Při řízení s polovičním krokem vlastně střídáme jednofázové a dvoufázové řízení. Nejbližší rotorový zub je nejprve přitahován jednou cívkou, pak jsou sepnuty obě zároveň a zub je mezi nimi a poté pouze druhá cívka. Tímto dosáhneme dvojnásobného rozlišení motoru.

Mikrokrokováním se dá dosáhnout velkého rozlišení pohybu. Princip mikrokrokování je v podstatě obdobný jako u řízení s polovičním krokem, avšak při mikrokrokování jsme schopni zastavit zub rotoru mezi dvěma cívkami ve více různých polohách. Při řízení s polovičním krokem je proud vždy buď 100% v jedné cívce, 100% v obou nebo 100% ve druhé. Mikrokrokování ovšem dosáhneme tak, že do jednotlivých cívek budeme pouštět poměrné zlomky proudu a tím pádem se nám rotor zastaví v několika mezipolohách mezi cívkami. Počet takovýchto mezipoloh je závislý na zvoleném mikrokrokování (v současné době se používá až 128 mikrokroků v rámci jednoho plného kroku motoru). Mikrokrokováním docílíme jednak zvýšené přesnosti polohování rotoru a také zmírnění mechanických rezonancí obzvláště při pomalém chodu motoru.

2.6 Výpočet a výběr vhodného motoru

Při navrhování vhodného motoru pro svoji CNC frézku jsem musel spočítat a určit několik základních parametrů pohonu. Mezi ně patřily hlavně potřebný krouticí moment motoru, rychlost, jakou se chci pohybovat a zrychlení (náběžná a sestupná rampa).

Nejprve jsem zvažil a změřil jednotlivé části CNC frézky. Výsledné hmotnosti a rozměry jednotlivých komponent jsou uvedeny v tabulce 2.1.

Tabulka 2.1: Změřené rozměry a hmotnosti jednotlivých součástí CNC frézky

Stoupání kul. Šroubu	0,005 m
Hmotnost kul. Šroubu – 100 cm	1,6 Kg
Hmotnost kul. Šroubu – 60 cm	1 Kg
Hmotnost kul. Šroubu – 30 cm	0,5 Kg
Hmotnost vřetena	2 Kg
Hmotnost portálu	29,3 Kg
Hmotnost „y“ sestavy	10 Kg
Hmotnost „z“ sestavy	3 Kg
Hmotnost pružné spojky	0,02 Kg
Poloměr pružné spojky	0,0125 m

Následně jsem si určil rychlost, kterou se chci pohybovat při rychloposuvu (přejezd frézky bez zátěže). Tuto rychlost jsem určil s ohledem na stoupání kuličkového šroubu a maximální možné otáčky krokového motoru bez použití převodu. Z datasheetu výrobce krokových motorů (microcon) jsem zvolil vhodný typ krokového motoru a následně díky maximálním otáčkám motoru, při kterých motor dosahuje krouticího momentu alespoň 1Nm (10 otáček/s = 600 RPM) určil maximální rychlost posuvu naprázdno. 10 otáček za sekundu při stoupání závitu 5

mm na otočku odpovídá maximální rychlosti pohybu naprázdno $5 \text{ cm/s} = 3 \text{ m/min}$.

Nyní jsem mohl přejít k počítání potřebného krouticího momentu motoru. Potřebný moment se skládá ze součtu statického a dynamického momentu potřebného pro pohyb danou osou a navíc z momentu, který vzniká na hřídeli při samotném frézování materiálu. Výpočet jsem po konzultaci s několika strojaři a odborníky z oboru provedl ve dvou fázích, dvěma způsoby.

Nejprve jsem si za pomoci měření a následného výpočtu určil statický moment, potřebný pro udržení frézky v ustáleném pohybu konstantní (maximální) rychlostí. Měření jsem provedl tak, že jsem na pružnou spojku, která spojuje motor s kuličkovým šroubem dané osy, namotal provázek, na který jsem postupně zavěšoval závaží o různé hmotnosti. Postupným zvyšováním hmotnosti závaží jsem došel na takovou hmotnost, při které se daná osa dala do pohybu a následně jsem přes průměr spojky, hmotnost závaží a tíhové zrychlení dopočítal krouticí moment, který byl potřebný pro takovýto pohyb osy.

Pro osu X jsem potřeboval závaží o hmotnosti $2,4 \text{ kg}$, které zajistilo plynulý pohyb celého portálu rychlostí zhruba 5 cm/s . Síla, kterou jsem tahal za provázek, tedy byla:

$$F_x = m_x \cdot g = 2,4 \cdot 9,81 = 23,544 \text{ N} \quad (1)$$

Krouticí moment na hřídeli motoru při použití pružné spojky o průměru $r=0,0125\text{m}$ tedy byl:

$$M_x = F_x \cdot r = 23,544 \cdot 0,0125 = \mathbf{0,2943 \text{ Nm}} \quad (2)$$

Dále jsem potřeboval určit dynamický moment, který na hřídel působí při rozběhu a zastavování stroje. Určil jsem si, že chci, aby se na maximální rychlost (5cm/s) stroj rozpohyboval za 2s . Zrychlení je v takovém případě $2,5\text{cm/s}^2 = 0,025\text{m/s}^2$.

Dle zákona zachování momentu hybnosti platí:

$$M \cdot \omega = F \cdot v \quad \rightarrow \quad M = F \cdot \frac{v}{\omega} \quad (3)$$

Kde $\frac{v}{\omega}$ značí převod (stoupání kuličkového šroubu). Stoupání kul. Šroubu v základních jednotkách SI můžeme rovněž vyjádřit jako:

$$s = \frac{v}{\omega} = \frac{l}{\varphi} = \frac{0,005}{2 \cdot \pi} \text{ [m/rad]} \quad (4)$$

Dynamický krouticí moment jsem dále počítal dle vztahu:

$$M_d = J_{celk.} \cdot \varepsilon \text{ [Nm]} \quad (5)$$

Příčemž úhlové zrychlení:

$$\varepsilon = \frac{2 \cdot \pi \cdot a}{s} = \frac{2 \cdot \pi \cdot 0,025}{0,005} = \mathbf{31,41 \text{ [rad/s}^2\text{]}} \quad (6)$$

Celkový moment setrvačnosti působící na hřídel motoru:

$$J_{celk.} = J_{portálu} + J_{šroubu} + J_{spojky} + J_{motoru} \text{ [kg} \cdot \text{m}^2\text{]} \quad (7)$$

Moment setrvačnosti portálu:

$$\frac{1}{2}m \cdot v^2 = \frac{1}{2}J \cdot \omega^2 \quad (8)$$

$$J_{portálu} = m_{portálu} \cdot \frac{v^2}{\omega^2} = 36 \cdot s^2 = 36 \cdot \left(\frac{0,005}{2 \cdot \pi}\right)^2 = 2,28 \cdot 10^{-5} \text{ [kg} \cdot \text{m}^2] \quad (9)$$

Pro výpočet momentu setrvačnosti kuličkového šroubu a pružné spojky jsem si tyto komponenty nahradil válcem o odpovídajícím poloměru (šroub - $r=8\text{mm}$, spojka - $r = 12,5\text{mm}$).

$$J_{šroubu} = \frac{m \cdot r^2}{2} = \frac{1,6 \cdot 0,008^2}{2} = 5,12 \cdot 10^{-5} \text{ [kg} \cdot \text{m}^2] \quad (10)$$

$$J_{spojky} = \frac{m \cdot r^2}{2} = \frac{0,02 \cdot 0,0125^2}{2} = 1,56 \cdot 10^{-6} \text{ [kg} \cdot \text{m}^2] \quad (11)$$

Moment setrvačnosti motoru (rotoru) jsem našel přímo v Datasheetu výrobce.

$$J_{motoru} = 1,2 \cdot 10^{-4} \text{ [kg} \cdot \text{m}^2] \quad (12)$$

Celkový moment setrvačnosti na hřídeli motoru je tedy:

$$J_{celk.} = J_{portálu} + J_{spojky} + J_{motoru} + J_{šroubu} = 2,28 \cdot 10^{-5} + 5,12 \cdot 10^{-5} + 1,56 \cdot 10^{-6} + 1,2 \cdot 10^{-4} = \mathbf{1,9556 \cdot 10^{-4} \text{ [kg} \cdot \text{m}^2]} \quad (13)$$

Dynamický moment potřebný pro rozjezd a zastavení portálu je tedy:

$$\mathbf{M_d} = J_{celk.} \cdot \varepsilon = 1,9556 \cdot 10^{-4} \cdot 31,41 = \mathbf{6,14 \cdot 10^{-3} \text{ [Nm]}} \quad (14)$$

Posledním důležitým údajem je určení momentu, který vznikne na hřídeli při frézování. Nejprve jsem si musel vypočítat/určit řezné síly, které vzniknou při frézování. Jelikož je takový výpočet náročný, závislý na mnoha parametrech a není předmětem mé práce ani mého oboru, inspiroval jsem se v diplomové práci Bc. Romana Hájka viz. [12]. Ten při použití dvoubřité frézy JABRO SOLID JS512 o průměru 8mm určil, že řezná síla vznikající při frézování a vrtání nelegované uhlíkové oceli s obsahem uhlíku nižším než 0,25%, s hloubkou záběru 5mm, rychlostí posuvu při frézování 764 mm/min a otáčkách nástroje při frézování 6366 ot./min je 302,4 N. Dále je v této práci počítán i minimální potřebný moment motoru při frézování, který je odvislý od řezné síly při frézování, stoupání závitu kuličkového šroubu (je zde použit stejný kuličkový šroub, který používám i já) a celkové účinnosti. Výsledná hodnota tohoto momentu je uvedena 0,29 Nm. Jelikož na svém stroji budu opracovávat především dřevo, případně hliník a jedná se mi pouze o orientační výpočet potřebného krouticího momentu motorů, jelikož je plánuji nakoupit několiknásobně předdimenzované, tak jsem se s touto hodnotou a celkovým postupem výpočtu spokojil.

Celkový minimální potřebný krouticí moment motoru je potom dán součtem statického, dynamického a řezného momentu.

$$\mathbf{M_{celk.}} = M_x + M_d + M_{řez.} = 0,2943 + 6,14 \cdot 10^{-3} + 0,29 \cong \mathbf{0,59 \text{ [Nm]}} \quad (15)$$

Výpočet potřebného krouticího momentu motorů uvádím pouze pro největší osu x. Zde bude potřeba motor nejsilnější, avšak kvůli jednoduchosti jsem se rozhodl koupit pro všechny osy motory stejné. Díky tomu vím, že pokud motor bude vhodný pro tuto osu, bude vhodný i pro ostatní dvě.

Vzhledem k povaze výpočtu a určení výsledné hodnoty potřebného momentu jsem se rozhodl kupované motory několikanásobně předdimenzovat. Takhle jsem se rozhodl i proto, že plánuji systém bez zpětné vazby od krokových motorů, takže musím zajistit, aby nikdy nedošlo k jejich přetížení a tím pádem ke ztrátě kroku. V technické praxi je běžné, že vybrané motory by měly být minimálně dvojnásobně momentově předdimenzovány při daných otáčkách.

Rozhodl jsem se pro koupi motorů SX24-2341 4,1 Nm od české firmy Microcon. Zajistím tím dostatečný výkon motorů v jakémkoliv případě a budu mít dostatečnou výkonovou rezervu i do budoucna, pro případné odladění a zvýšení rychlosti provozu. Tyto motory jsou schopné rychlosti až 1200 ot./s, což při stoupání kuličkového šroubu 5mm na otočku odpovídá rychlosti pohybu až 10 cm/s (6m/min). Mám tedy do budoucna dostatečný prostor na zrychlení a odladění chodu celého stroje v praxi, jakmile bude vše zapojeno. Motory zároveň mají dostatečný krouticí moment v nízkých otáčkách a tím je zajištěno, že určitě nebude docházet ke ztrátě kroku při frézování.

3 ŘÍDICÍ ELEKTRONIKA

Řídicí elektronikou frézky je myšlena veškerá elektronika a hardware mezi uživatelem a fyzickým strojem (jeho motory). Řídicí elektronika zajišťuje, že požadavek, který uživatel na stroj vyvine, se převede na skutečné pohyby stroje.

Existuje spousta možností a konfigurací, jak motory stroje rozpohybovat, z nichž nejčastěji používané v hobby či semi-profi sféře jsou:

- PC/raspberry Pi/beaglebone... + breakoutboard + drivery
- PC + HW interpolátor (smoothstepper) + breakout board + drivery
- PC + smoothieboard
- PLC + (drivery)

Hlavní HW komponenty pro řízení CNC frézky, které jsou v nějaké formě přítomné v každé možnosti řízení, jsou:

- Počítač
- Breakout board
- Drivery krokových motorů
- Zdroj napájení

3.1 Počítač

Pod pojmem počítač je myšleno zařízení, které přímo interaguje s uživatelem, do kterého uživatel zadává vstupní data, a následně díky potřebným programům jsou tato data zpracována. Takovýto počítač může reprezentovat například stolní PC, na kterém jsou nainstalovány potřebné programy, malý kompaktní levný počítač, jako například Raspberry Pi, BeagleBone Black, Arduino, atd. nebo průmyslové řešení v podobě PLC. Každá z těchto možností s sebou nese své výhody, ale i nevýhody.

Pozici nejčastěji používaného prostředku pro řízení hobby CNC frézky zastává nejspíše PC. Spousta stavitelů se rozhoduje právě mezi použitím PC a malého kompaktního počítače v podobě Raspberry Pi, Arduina, atd. V obou případech jsou nutné další komponenty pro běh celého systému a počítač zde plní pouze funkci výpočetního ústředí, kde běží požadované programy. Jedná se hlavně interpret G-kódu, případně navíc kompletní struktura programů, které z grafického návrhu obrobku tvoří G-kód. Důvodem proč velká spousta lidí volí právě PC je, že snad každý kutil má doma jedno či dvě staré PC, které může v projektu použít a nemusí tak utrácet peníze za nic nového. Toto je i můj případ, a proto jsem se rozhodl pro svou stavbu zvolit jeden ze stolních počítačů, který se mi nevyužitý válí doma, jako výpočetní centrum. Největší nevýhodou použití stolního počítače je, že při opracovávání obrobku dochází k prášení a stolní počítače obecně

nejsou konstruovány na to, aby pracovaly v podmínkách, kde se frézka bude používat (na dílně, v prašném prostředí, ...). Tuto nevýhodu ovšem lze lehce kompenzovat tím, že počítač bude umístěn ve větší vzdálenosti od samotného stroje.

Pokud někdo doma nemá po ruce starý stolní počítač, nebo chce stroj udělat kompaktnější a odolnější, zřejmě sáhne po variantě s Raspberry Pi, Arduinem či BeagleBone Black. Základní podstata těchto kompaktních, modulárních počítačů je stejná. Vše potřebné se nachází na jednom tištěném spoji zhruba o velikosti pánské peněženky a pořizovací cena je velmi příznivá. Programy a operační systém jsou instalovány na microSD kartu. Malý kompaktní počítač je velmi jednoduché zapouzdřit tak, aby se do něj nedostal prach a nečistoty a tak může být umístěn přímo na stroji. Navíc díky velké řadě vstupů a výstupů je možné k němu připojit i dotykové LCD a ovládat celý systém frézky pomocí něj. Není zde tedy nutnost dedikované stanice, kde bude umístěn klasický stolní počítač s monitorem, nebo laptop.

Poslední variantou je použití PLC. Tato varianta je zcela jistě nejlepší, jelikož PLC jsou konstruována přesně pro takovéto účely. Zároveň ne vždy je zde nutnost používat dodatečné komponenty, jako breakout board, či drivery motorů, jelikož PLC už mohou být vybavena dostatkem vstupů a výstupů. PLC se ovšem pro hobby CNC frézky prakticky nevyužívá, díky vysoké pořizovací ceně, která při stavbě hobby CNC frézky může převýšit celkovou hodnotu všeho ostatního na stroji dohromady

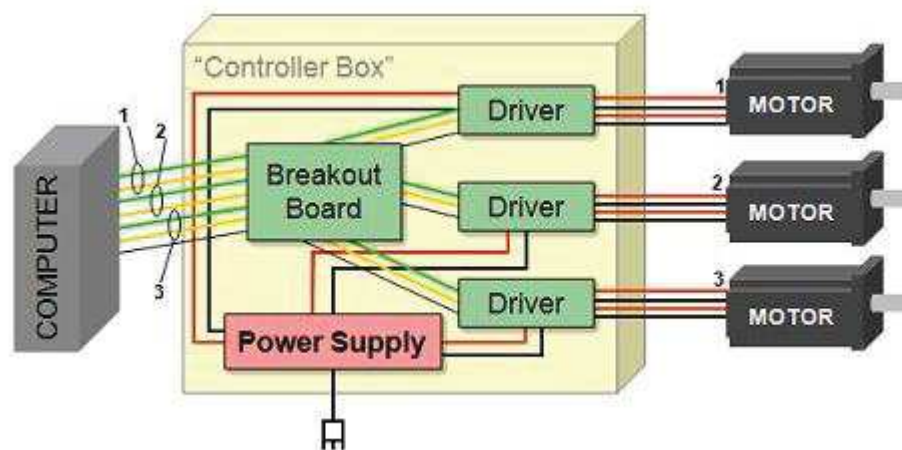
3.2 CNC kontrolér

Obvykle reprezentuje kompletní systém ovládání stroje. Tento systém může obsahovat ochranné obvody, drivery motorů, zdroj napájení, limitní a další spínače a periferie.

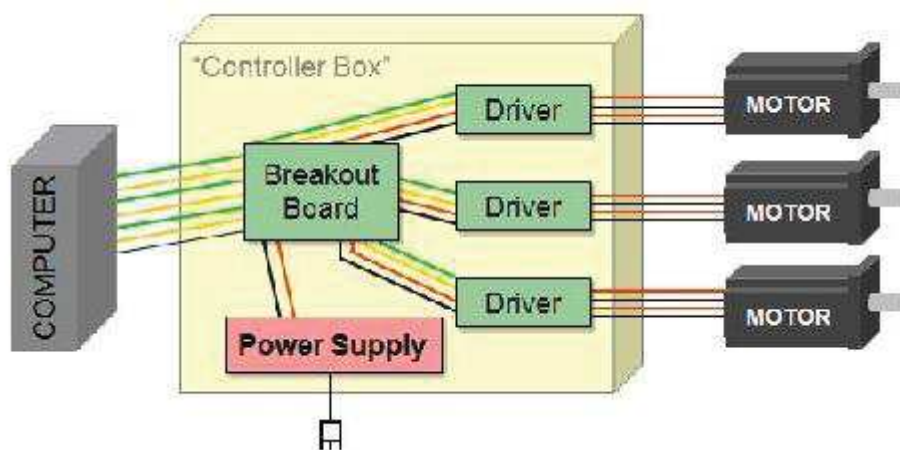
Napájecí zdroj napájí drivery a ostatní periferie, které potřebují externí napájení.

Ochranné obvody obsahuje breakout board, který distribuuje signály driverům a zároveň umožňuje jednoduché připojení dalších periférií, jako jsou např. limitní spínače. Součástí jsou i pojistky, které chrání proti přepětí a dalším nechtěným vlivům.

Drivery motorů dostávají signály, na základě nichž generují pulsy o požadovaném proudu a napětí, které vyvolají pohyb motorů.



Obrázek 3.1: Schéma řízení CNC frézky, kde je každý driver KM napájen samostatně ze zdroje [14]



Obrázek 3.2: Schéma řízení CNC frézky, kde jsou drivery napájeny pomocí breakout boardu, který slouží jako rozvod napájení ze zdroje [14]

3.2.1 Breakout board

Deska, která rozděljuje (break out) piny např. z LPT kabelu tak, aby byly přístupné např. pro drivery motorů. Breakout board může rozvádět jen signály (obr. 3.1), nebo i napájení (obr. 3.2) a může, ale i nemusí pro svou funkci požadovat napájení. Pokud požaduje napájení, je to většinou proto, že zároveň slouží jako ochranný obvod např. proti přepětí a to pomocí optospínačů. Breakout board je k počítači nejčastěji připojen buď pomocí LPT kabelu, nebo pomocí USB. Nelze říct, že jedna varianta je výrazně lepší než druhá, proto volba připojení záleží hlavně na tom, jaké porty nabízí použitý počítač. [14]

3.2.2 Drivery krokových motorů

Drivery motorů dostávají signály, na základě nichž generují pulsy o požadovaném proudu a napětí, které vyvolají pohyb motorů.

Driver potřebuje alespoň 4 spojení, aby mohl fungovat. [15]

1. Direction signál
2. Step signal = Kolik má udělat kroků
3. Kladné napětí ze zdroje napětí
4. Zem ze zdroje napětí

Driver interpretuje STEP a DIR instrukce z počítače a nabíjí požadované cívky tak, aby se motor otáčel v požadovaném směru o požadovaný počet kroků. [15] Velikost proudu přiváděného na cívky motoru určuje jeho krouticí moment.

Klíčové parametry při výběru vhodného driveru krokových motorů:

a. Počet os

Většina hobby strojů má 3 drivery a breakout board, kvůli jednoduchosti provedení. Nicméně je zapotřebí se ujistit, že taková sestava zvládne i další potřebná připojení, jako jsou např. limitní spínače, atd. Existují i komplikovanější a složitější možnosti, jako například mít breakout board a driver pro každou osu samostatně, nebo existují drivery pro více os, každopádně vždy se jedná o dražší a většinou i složitější provedení.

b. Unipolární vs. Bipolární

Tato volba se týká se pwm driverů. Unipolární drivery se přestávají používat. Samotný KM může být definován jako unipolární či bipolární, to odkazuje na typ vinutí pro nejpoužívanější (2 fázové) KM. Rozdíl mezi unipolárním a bipolárním řízením KM je popsán v kapitole: 2.5.1.1

Pokud není specifikováno, zda je driver unipolární či bipolární, je to možné určit pomocí počtu portů pro připojení motorů. Bipolární driver má pouze 4 porty, zatímco unipolární driver bude mít víc než 4. Unipolární driver je kompatibilní s 5,6 nebo 8 drátovými KM. KM se 4 dráty jsou bipolární a nejsou kompatibilní s unipolárními drivery.

c. Výstupní proud

Driver by měl mít vždy uvedený výstupní proud, většinou udáván na fázi. Zároveň by měl uvádět minimální napětí na fázi. Jedná se o jeden z nejdůležitějších parametrů, protože určuje, jak výkonný motor můžeme s driverem použít.

d. Módy krokování

Plný krok/poloviční krok/mikrokrokování. viz. kapitola: 2.5.1.3

e. Typ driveru [15]

1. L/R Drivery

Driver dodává konstantní napětí různým cívkám k dosažení otáčení. Proud v každé cívce je závislý na indukčnosti (L) a rezistenci (R) motoru. Jedná se o typ driveru, který se příliš nepoužívá, protože není moc efektivní.

2. Bi-level drivery

Driver používá zdroj vysokého napětí, aby rapidně zrychlil nárůst proudu na cívkách motoru. Pak je vysoké napětí odpojeno a zdroj nízkého napětí udržuje konstantní úroveň. Tyto drivery jsou velice efektivní a vyskytují se u high-end strojů. Méně se zahřívají motory, ale není zde možné mikrokrokování, pouze poloviční kroky a plný krok.

3. Chopper (PWM) drivery

Nejpopulárnější drivery KM. Známé i jako driver s konstantním proudem, protože dodává konstantní proud cívkám. Každé cívce je dodáno velice vysoké napětí a poté useknuto (chopped) když je dosažen požadovaný proud. Tyto drivery vyžadují pouze jeden zdroj napětí (proudu). Velikost proudu na každou cívku se dá většinou měnit např. potenciometrem. Tyto drivery podporují mikrokrokování, což vede k hladšímu průběhu a většímu rozlišení.

f. Rychlost přepínání (pouze PWM)

Rychlost přepínání je frekvence, na které PWM driver operuje. Tato hodnota by měla být minimálně 20kHz. Čím je tato hodnota vyšší, tím plynuleji bude driver pracovat.

g. Frekvence krokování (step frequency)

Maximální vstupní frekvence pro příkazový signál. Dobrý driver bude mít vstupní frekvenci 0 – 200kHz. Toto je důležité, pokud se snažíme dosáhnout vysoké rychlosti a přesnosti.

h. Ochranné obvody

Většina driverů nabízí vnitřní ochranné obvody pro ochranu motorů a generátoru signálu. To je dosaženo za pomoci optospínačů a pojistek. Pokud driver nenabízí ochranu, je potřeba se ujistit, že breakout board ano.

i. Standby mód

Některé drivery nabízí standby mód, který snižuje napětí dodávané každému motoru po určité době, co jsou motory nečinné. To pomáhá redukovat teplo vytvářené motory, což zvyšuje jejich životnost a efektivitu.

3.3 Další HW komponenty

Tyto komponenty se mohou, ale nutně nemusí, použít při stavbě CNC frézky. Jedná se hlavně o takové komponenty, které buď hardwarově nahrazují nějakou činnost, která je jinak dělána softwarově, nebo spojují několik již zmíněných komponent do jedné, kvůli jednoduchosti zapojení a instalace. Mám na mysli hlavně HW interpolátory a smoothieboard.

3.3.1 Interpolátor

Máme zadaný pohyb z bodu A do bodu B. Takovýto pohyb může být buď po přímce, nebo po nějaké kruhové trajektorii. Frézka se ovšem pohybuje jen přímočaře a to v ose x, y a z. Pokud se tedy chceme dostat z bodu A do bodu B, musíme si dráhu rozdělit na malé úseky pohybu v ose x, y a z. Správně rozdělení dráhy do takovýchto dílčích elementů pro danou osu tak, aby byl pohyb co nejplynulejší a nejpřesněji kopíroval ideální požadovanou trajektorii, je úlohou právě interpolátoru.

Existují dva typy interpolace podle zadaného pohybu a to interpolace lineární, pro přímočarý pohyb a interpolace kruhová, pro pohyb po kruhovém oblouku. Interpolaci pohybu je potřeba udělat pro každý pohyb, který zahrnuje použití více než jedné osy frézky. Jednodušší a levnější varianta je interpolace softwarová, kdy interpolaci řeší řídicí software, nainstalovaný na používaném počítači. Pokud nám jde ovšem o zvýšení rychlosti a přesnosti provozu frézky, může interpolaci pohybu provádět i samostatná hw komponenta, HW interpolátor. Příkladem takového HW interpolátoru je SmoothStepper. Jedná se o 6-osý HW interpolátor pro řídicí software MACH3. Smoothstepper produkuje řídicí pulsy pro krokové motory až 80 krát rychleji, než při použití LPT portu a generování pulsů čistě programem MACH3. Zároveň jsou pulsy kvalitnější a lépe časované. HW interpolátory zároveň ulevují řídicímu programu a tím pádem i počítači od nutnosti provádět potřebné výpočty pro generování těchto pulsů, čímž výrazně snižují vytížení procesoru.

3.3.2 Smoothieboard

Smoothieboard je kontrolér, který běží na Smoothieware firmware. Smoothieware je opensource interpret G-kódu. Smoothieboard je kontrolér, který v sobě spojuje funkce hned několika výše zmíněných prvků. Smoothieboard v2 v první řadě nabízí Cortex-M4 čip s 8MB flash paměti a 264kB RAM. Díky tomuto procesoru je deska mnohem výkonnější, než srovnatelná zařízení, pracující s ARM procesory, a tím pádem je možné dosáhnout vyšších rychlostí a přesností stroje a připojit více prvků včetně dotykové obrazovky. Deska dále obsahuje všechny potřebné vstupní a výstupní porty pro připojení koncových spínačů, ovládání vřetena, atd. takže slouží jednak jako breakout board a zároveň obsahuje hned několik driverů pro krokové motory s možností mikrokrokování. Drivery se liší v závislosti na verzi smoothieboardu. Smoothieboard v2 obsahuje drivery, které jsou schopny napájet motory až 30V, 2,8A a umožňují až 1/128 mikrokrokování. Pokud by parametry integrovaných driverů nestačily, nachází se na desce porty pro připojení externích driverů pro případ, že chceme touto deskou ovládat i větší stroje. Je zde slot na microSD kartu, díky níž je možné nahrávat soubory na zpracování stejně jako konfigurační a aktualizací soubory. Spojení s PC je možné buď pomocí USB, nebo ethernet kabelu.

Smoothieboard se hodí hlavně na použití v 3D tiskárnách, ale tuto desku je možné také použít i na ovládání dalších CNC strojů, jako například laserových řezaček a CNC frézek. Bohužel v současnosti je dostupná pouze verze 1, Smoothieboard v2 je stále ve vývoji a datum uvedení do prodeje zatím není známo. Další z nevýhod je i to, že aktuálně dostupná verze 1 stojí cca 4500 Kč bez dopravy.

4 SOFTWARE

Software pro řízení CNC frézky se dělí do několika kategorií. Nejprve se vytvoří grafický návrh obrobku v nějakém CAD programu. Grafický návrh obrobku je poté potřeba zpracovat a převést ho na G-kód. G-kód je programovací jazyk, pomocí kterého se řídí NC a CNC stroje. Jedná se v podstatě o sadu instrukcí/příkazů, které následně interpret G-kódu převádí na signály, díky nimž dochází k požadovaným pohybům jednotlivých os.

V postupu interpretace grafického návrhu obrobku na výsledné pohyby stroje je zapotřebí několika programů.

4.1.1 CAD software

CAD = computer aided design (počítačem podporované projektování). CAD programy slouží k vytvoření grafického návrhu výsledného obrobku. Takovýto návrh může obecně být buď 2D, nebo 3D, avšak pro vytváření obrobků na CNC frézce se používají 3D návrhy. CAD programy exportují návrhy do souborů *.dxf nebo *.svg, které jsou následně zpracovány CAM programy. Některé CAD programy mohou mít dodatečně nainstalované pluginy, díky kterým lze rovnou generovat dráhy nástroje. Mezi v současnosti nejpoužívanější CAM programy pro návrh obrobků pro CNC frézky patří například SolidWorks, SketchUp, FreeCAD, Inventor, atd. Existuje nespočet dalších specializovaných programů, jejichž výčet a popis vlastností je nad rámec této práce. SolidWorks nabízí nejlepší možnosti nastavení při návrhu, avšak se jedná o placený program, jehož pořizovací cena se pohybuje v řádu desítek tisíc korun v závislosti na zvolené verzi programu. Na druhou stranu SketchUp je CAD program od společnosti Google, který nabízí pro běžné hobby použití dostatečné možnosti návrhu, je velmi intuitivní a jednoduchý na naučení a je zcela zdarma. Proto jsem se taky rozhodl při své stavbě pracovat s tímto CAD softwarem.

4.1.2 CAM software

CAM = computer aided manufacturing (počítačem podporovaná výroba). CAM program používaný při ovládní CNC frézky umožňuje převést nákres součástky vytvořené v CAD programu na G-kód. G-kód je programovací jazyk, sloužící pro programování a ovládní NC a CNC strojů. CAM program tedy přímo generuje cestu nástroje (tool path), kterou musí v tomto případě fréza vykonat při obrábění materiálu a následně z této cesty nástroje generuje G-kód (tomuto kroku říkáme Post processing). Pro jednodušší operace jako například rovné čáry, kružnice a

obecně jednoduché objekty můžeme rovněž G-kód napsat ručně. Opět existuje několik volně dostupných i placených CAM programů na různé úrovni, jejichž kompletní rozbor a popis je mimo rozsah této bakalářské práce. Nutno podotknout, že CAM programy jsou nejhůře dostupná skupina programů, potřebných k operování CNC. Volně dostupných CAM programů není mnoho. Mezi nejlepší z volně dostupných patří například programy Fusion 360 (zdarma pro studenty), PyCAM, nebo plugin pro Google SketchUp – SketchUCam. Programy s více funkcemi již bývají placené. Z těch levnějších je to například ESTLCam, MeshCAM, nebo SpurtCAM a na profesionální úrovni najdeme například oblíbený program MasterCAM.

4.1.3 Interpret G-kódu – ovládací software pro CNC frézku

Jedná se o poslední z řady programů potřebných pro ovládání a programování CNC frézky. CAM program vygeneruje soubor s G-kódem, avšak teprve interpret G-kódu je program, který frézku na základě těchto příkazů rozpohybuje. Zde již není tak velké množství programů, které jsou užívány. Nejčastěji se používají z placených programů Mach3 a z volně dostupných LinuxCNC (dříve označován jako EMC2).

Velkou výhodou placeného programu Mach3 je, že je mnohem jednodušší na nastavení i obsluhu. Grafické rozhraní je příjemnější a nepotřebuje nastavovat tak velké množství parametrů jako u LinuxCNC. Cena jedné licence programu Mach 3 je v současnosti 175 dolarů. Mach3 běží na systémech Windows a díky nízkým nárokům ho lze nainstalovat a provozovat i na starém a levném počítači.

Na rozdíl od Mach3 je LinuxCNC volně dostupný a běží na systémech Linux. Pokud chceme pomocí LinuxCNC frézku ovládat, je ovšem nutné nejprve nainstalovat real-time (RTAI) verzi jádra operačního systému. RTAI kernel pro Linux je dostupný pouze pro zastaralé distribuce Linuxu (např. Ubuntu Precise, nebo Debian Wheezy). Návod na instalaci RTAI jádra lze nalézt přímo na stránkách projektu LinuxCNC (viz. [18]). LinuxCNC nabízí větší možnosti nastavení, programování a ovládání než Mach3. Díky tomu, že se jedná o open source projekt a kompletní zdrojový kód je dostupný, není problém si LinuxCNC upravit dle své libosti, dopsat si své vlastní části, řídicí nebo i vizualizační komponenty. Další výhodou LinuxCNC oproti Mach3, které se ovšem v poslední době a při použití správného HW začíná smazávat je i fakt, že Mach3 občas může být pomalý a ztrácet krok, což se u LinuxCNC nestává. Obecně se tedy dá říct, že kdo neplánuje příliš měnit či upravovat ovládací systém frézky a chce jen co nejdříve začít vyrábět, může sáhnout po programu Mach3. Pro složitější úpravy, lepší možnost odladění, rychlejší chod a v neposlední řadě ušetřené peníze, je lepší volbou LinuxCNC.

Zatím jsem zmínil dva hlavní programy pro interpretaci G-kódu. Existuje jich ovšem opět více, přičemž poslední, který zmíním je MachineKit. MachineKit vznikl odštěpením z projektu LinuxCNC v roce 2014. Oproti LinuxCNC se MachineKit snaží portovat tento SW na co nejširší škálu HW platforem, dosáhnout nezávislosti na x86 platformě a RTAI jádru. Projekt se momentálně pořád vyvíjí a zatím není dostatečná uživatelská základna, či komunita, která by ho podporovala. MachineKit pracuje úspěšně například na platformě BeagleBone Black.

5 POUŽITÉ ZAPOJENÍ

5.1 Řídicí elektronika

Ve své konstrukci jsem použil zapojení řídicí elektroniky dle schématu v příloze A. Pro generování STEP/DIR signálů využívám starý stolní PC, který je pomocí LPT kabelu připojen k breakout boardu. Ten je z PC rovněž napájen, a to pomocí USB kabelu. Breakout board dále distribuuje signály na drivery KM, ke kterým jsou motory připojeny bipolárně sériově. K breakout boardu je navíc připojeno celkem 6 koncových spínačů, které zajišťují, že nedojde při špatně napsaném G-kódu k poškození stroje. Drivery jsou napájeny každý samostatně ze spínaného zdroje, který je schopný poskytnout až 50V a 16A. Indikaci správného napájení driverů zajišťuje vestavěná LED dioda. Napětí do zdroje a zároveň na vřeteno je vedeno přes kolébkový spínač. Zdroj pro drivery je tak jednodušší zapínat a vypínat a zároveň se jedná o další bezpečnostní faktor, kdy při potřebě náhlého vypnutí je takto odstaveno napájení vřetene i motorů současně, i když vřeteno na řídicí systém CNC frézky není jinak napojeno. Díky takovému opatření se jedná o lepší bezpečnostní zajištění než použití E-STOP tlačítka, které by, v případě nutnosti použití v tomto zapojení, odstavilo pouze motory, ale nevypnulo by vřeteno.

Použité komponenty a jejich parametry:

Breakout board HY-JK02-M

- Připojení až 5-ti osého stroje (5 driverů – signály STEP, DIR a ENABLE)
- Propojení s PC pomocí paralelního portu
- Napájení pomocí USB nebo přímé připojení DC 5V
- Pro každou osu je možné připojit koncové spínače
- Možnost připojení E-stop tlačítka, vřetene a externího displeje
- Port pro externí manuální ovládání

Drivery Leadshine M542

- Napájecí napětí až 50V DC
- Výstupní proud až 4,2 A (3,0 A RMS)
- Mikrokrokování – 15 volitelných rozlišení – až 1/128
- Opticky izolovaný vstup - ochrana proti zkratu, přepětí i velkému proudu
- Maximální frekvence přijímaných pulzů až 300 kHz

Motory Microcon sx24-2341

- Hybridní krokové motory
- Zapojeny bipolárně sériově
- NEMA 24
- Krouticí moment až 4,1 Nm
- Až 1000 RPM při krouticím momentu 1 Nm
- Momentová charakteristika použitých motorů viz. Obrázek 2.7

Spínaný zdroj

- Výstup 50V a 16 A (800W)
- Možnost nastavení výstupního napětí potenciometrem na zdroji
- Vstup 230V AC

Vřeteno DWT GS06-27 LV

- Příkon 600W
- Hmotnost 1,46 kg
- Manuálně regulovatelné otáčky v 6-ti stupních (12000-27000 min⁻¹)
- Upínací kleština 6mm

5.2 PC

Pro běh LinuxCNC a generování STEP/DIR signálů používám starý stolní počítač, který jsem měl doma.

Parametry PC jsou:

- Procesor: Intel Celeron 3,2 GHz
- RAM: 2x1GB DDRII 266 MHz
- Grafická karta: Asus NVIDIA GeForce 8500GT 512MB DDR2
- SSD 32 GB

Nejprve jsem si zvolil systém, na kterém chci, aby LinuxCNC běžel. Na výběr jsem měl ze dvou distribucí Linuxu, jelikož pokud chceme CNC frézku řídit pomocí LinuxCNC je potřeba mít nejprve nainstalovanou real-time verzi jádra systému (RTAI kernel). Zvolil jsem Ubuntu 12.04.5 LTS (Precise Pangolin), jelikož s Ubuntu jsem měl již drobné zkušenosti z minulosti a jedná se, dle mého názoru, o uživatelsky nejprístupnější distribuci Linuxu. Použití zastaralé verze Ubuntu 12.04.5 je z toho důvodu, že pro novější verze ještě není k dispozici RTAI kernel, nebo jeho instalace není dostatečně popsána. Z archivů Ubuntu jsem stáhnul a nainstaloval již zmíněnou verzi Ubuntu 12.04.5 LTS (Precise Pangolin) a následně

postupoval dle návodu na stránkách projektu LinuxCNC pro instalaci RTAI jádra a samotného LinuxCNC. Jedná se o relativně jednoduchý proces, který je popsán na stránkách projektu viz [18].

5.3 LinuxCNC

5.3.1 Latency test + výběr finálního PC

Obecně se dá říci, že LinuxCNC nepotřebuje extra výkonný HW pro svou správnou funkci. Je však důležité zajistit, aby pulzy, které PC vytváří, byly kvalitní a dostatečně rychle generované. Neplatí zde ovšem přímá úměra mezi hrubým výpočetním výkonem použitého HW a schopností generovat pulzy dostatečně rychle a kvalitně. Velice důležitým krokem je proto provedení Latency testu, který je možno spustit po instalaci LinuxCNC.

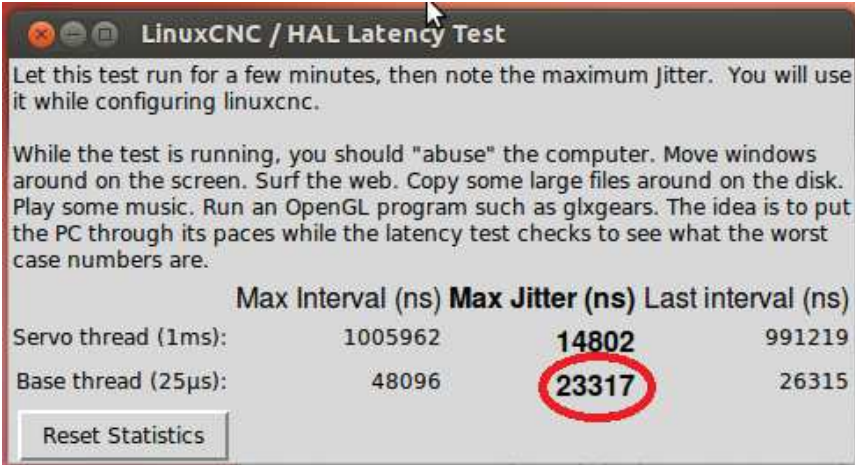
Tento test měří maximální odezvu procesoru na vyslaný požadavek. Jedná se v podstatě o to, jak dlouho počítač trvá, než přeruší to, co dělá a odpoví na externí požadavek. Proto je doporučeno při testování odezvy vytížit počítač co nejvíce (např. spuštěním videa, pohybováním okny nebo kopírováním velkých souborů na disku). Výsledek testu je následně směrodatný pro určení, jak rychle je počítač schopný generovat pulzy a tím pádem jak rychle se mohou motory otáčet.

Zde jsem narazil na první problém. PC, které jsem původně pro řízení zvolil (nazvěme ho „PC1“), mělo výsledek Latency testu – „Max Jitter“ cca 1 000 000 ns. Aby bylo možné jakékoliv PC úspěšně použít pro generování pulzů a vyhnuli jsme se tak použití externích HW interpolátorů, je potřeba, aby odezva byla co nejnižší (max. 50 000 ns). Díky tomu bylo „PC1“ naprosto nepoužitelné pro řízení.

Rozhodl jsem se tedy zakoupit výkonnější PC („PC2“ - Dell Optiplex 755 SFF s procesorem Intel Core 2 Duo E8400 3,0GHz a 4 GB DDR2 RAM). Nyní jsem po odladění dosahoval výsledků cca. 10 000ns, což bylo velmi dobré a pro mou aplikaci plně dostačující. Narazil jsem ovšem na nový problém, kdy i přes uspokojivé výsledky Latency testu jsem nebyl schopný motory otáčet. Docházelo v nich pouze k bzučení a cukáním, nepatrným pohybům, které nešlo eliminovat ani snížením akcelerace a maximální rychlosti. Na internetu existuje spousta návodů, jak snížit Latency systému na minimum, včetně úprav přerozdělování interruptů při použití vícejádrových procesorů tak, aby určitá přerušení (ta od CNC stroje) zpracovávalo pouze jedno jádro a druhé jádro všechna ostatní. Nastavil jsem tedy, aby se tato přerušení řešila pouze v jednom jádru a vše ostatní v druhém. Nic z použitých úprav mi nepomohlo a motory se stále neotáčely. Taktéž jsem vyzkoušel osadit do „PC2“ jiný procesor z řady Intel Core 2 Duo, se kterým mělo „PC2“ Latency cca. 50 000 ns a motory se opět netočily.

Nakonec jsem zkusil nainstalovat LinuxCNC na můj laptop s procesorem Intel Core i5-3320M 2,6 GHz a i přes výsledek Latency testu cca 50 000 ns, jsem byl s motory schopný plynule otáčet. Po cca týdenním neustálém ladění a zkoušení jsem vydedukoval, že chyba je pravděpodobně v řadě procesorů Intel Core 2 Duo. Latency test mi vždy vyšel více než dobře, celá konfigurace byla provedena správně, ale nebyl jsem schopný motory ani pootočit. Touto dobou jsem si zároveň uvědomil, že odezva systému nemusí vždy být přímo úměrná hrubému výpočetnímu výkonu použitého HW.

Rozhodl jsem se tedy upustit od použití „PC2“ a začal jsem zkoušet rozběhnout LinuxCNC na všech ostatních PC, které doma mám. Nakonec se mi povedlo rozjet LinuxCNC na jednom z počítačů, který jsem předem odsoudil jako málo výkonný. Konfigurace tohoto finálního PC je na začátku kapitoly 5.2. Nyní dosahuji výsledků Latency testu cca 25 000 ns. Není to tak dobré, jako při použití „PC2“, avšak nyní jsem schopný motory plynule otáčet rychlostí až 1500 mm/min. Původní požadavek na rychlost pojezdu bez zátěže byl až 3000 mm/min. Po kompletním zapojení stroje a testu v reálném provozu jsem zjistil, že mechanická konstrukce stroje je poddimenzovaná a dochází k rozvibrování osy Y i při těchto rychlostech (1500 mm/min). Nyní mám tedy rychlost sniženou na cca 500 mm/min. a pohyb je plynulý a bez sekání. Tím rovněž zajišťuji, že PC má dostatek času na generování pulzů a nedochází tedy ke ztrátě kroku vlivem přílišných nároků na rychlost.



	Max Interval (ns)	Max Jitter (ns)	Last interval (ns)
Servo thread (1ms):	1005962	14802	991219
Base thread (25µs):	48096	23317	26315

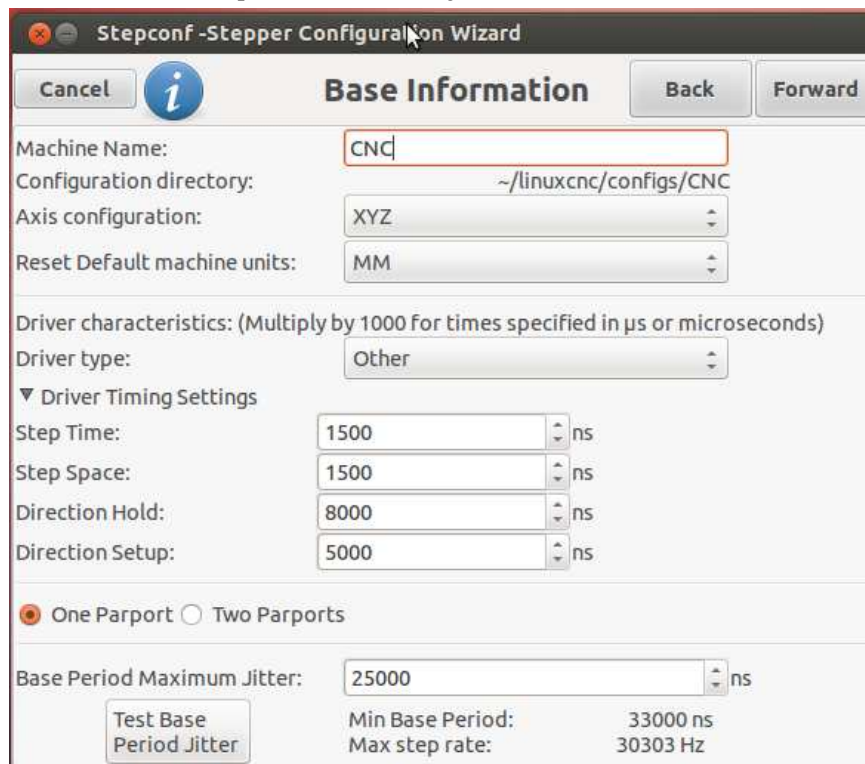
Obrázek 5.1 Výsledek Latency testu pro konečnou sestavu

5.3.2 Nastavení parametrů

Jakmile jsem LinuxCNC nainstaloval a odzkoušel pomocí Latency testu, že počítač, který používám je pro řízení vhodný, mohl jsem se věnovat nastavení programu. LinuxCNC obsahuje knihovny pro některé kompletní systémy, avšak v mém případě jsem musel nastavení provést manuálně dle HW, který používám. K nastavení všech potřebných parametrů slouží další z nástrojů LinuxCNC, a to je

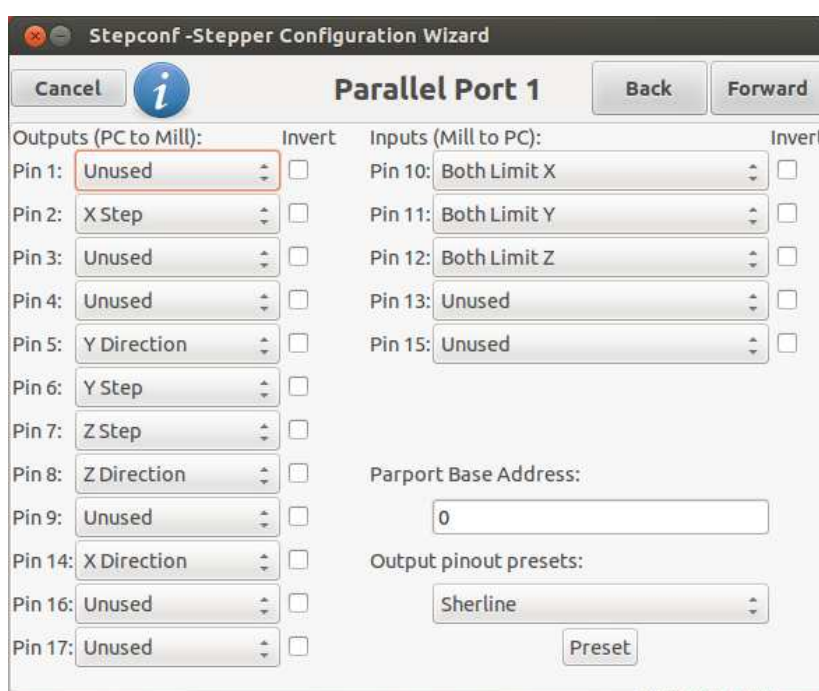
„stepconf wizard“. Zde jsem vytvořil novou konfiguraci. Na seznámení se s prostředím LinuxCNC je možné si vytvořit konfiguraci se simulovaným HW, kdy není ještě nutné mít připojeny veškeré komponenty. Tato možnost slouží pro prvotní seznámení se s prostředím LinuxuCNC, pokud ještě nemáme veškerý HW, který potřebujeme, což byl zpočátku i můj případ. Jakmile mi přišel veškerý potřebný HW, provedl jsem následující finální nastavení.

V prvním okně stepconf wizardu („Base configuration“) jsem nastavil obecné informace, jako typ stroje (počet os), jednotky, jméno stroje a hodnoty časování driverů („Driver Timing settings“). Hodnoty pro nastavení časování driveru můžeme zjistit buď z datasheetu driveru, nebo ze seznamu hodnot pro většinu běžně používaných driverů, který se nachází přímo na stránkách projektu LinuxCNC viz. [20]. V prvním okně je rovněž možné spustit již zmíněný Latency test. Jeho výsledkem je vyšší z hodnot „Max Jitter“ v nanosekundách, kterou, když byla vyhovující (<50 000 ns), jsem uvedl do příslušného políčka „Base Period Maximum Jitter“ a stepconf wizard sám vypočítal „Max step rate“. Dle hodnoty „Max step rate“ [Hz] jsem se poté řídil při nastavování maximální rychlosti pojezdů. Jedná se o teoretickou limitní frekvenci generování STEP/DIR pulzů. Schválně uvádím „teoretickou“, jelikož v praxi jsem musel při zkoušení posuvů nanečisto na stole používat nižší rychlosti, než mi povoloval tento faktor, jinak docházelo ke ztrátě kroku při maximální rychlosti.



Obrázek 5.2: LinuxCNC - Base configuration

Druhé okno stepconf wizardu („Parallel port 1“) je určené pro nastavení paralelního portu, respektive jeho pinů. Nastavil jsem zde tedy dle datasheetu mého breakout boardu (viz [21]) rozložení signálů dle jednotlivých pinů paralelního portu. Konkrétní použitou konfiguraci je možno vidět viz Obrázek 5.3. V tomto okně můžeme taktéž nastavit adresu paralelního portu („Parport Base Address“). Pokud necháme 0, tak je použita defaultní adresa 0x378 a zároveň je zde možnost přímo zvolit konkrétní rozložení pinů, pokud používáme jeden z breakout boardů, lépe řečeno jedno z rozložení, které poskytuje knihovna LinuxCNC. V mém případě je rozložení pinů jiné, než u nabízených možností a tak jsem musel hodnoty nastavit manuálně, dle datasheetu breakout boardu.



Obrázek 5.3: LinuxCNC - nastavení Paralelního portu

V dalším okně („Parallel port 2“) je možné nastavit rozložení pinů ještě pro druhý paralelní port, pokud jsme ovšem tuto možnost dříve zvolili a skutečně dva paralelní porty používáme. V opačném (i mém) případě se mi okno nezobrazí, jelikož si vystačím s jedním paralelním portem a piny co nabízí.

Nyní se již dostávám k samotnému nastavování parametrů jednotlivých os. Dle typu zvoleného stroje následuje několik oken, ve kterých nastavujeme parametry pro každou osu zvlášť. V mém případě se jedná o tři okna, pro tři osy.

Pro každou osu se nastavují následující informace:

- Počet kroků motoru na 1 otočku

(v mém případě 200 kroků na otočku u všech motorů).

- Mikrokrokování driveru

Experimentoval jsem s tímto nastavením hlavně po osazení motorů na stroj, a to z důvodu cukání při vyšších rychlostech, ale nakonec jsem u všech tří os ponechal nastavené mikrokrokování na hodnotě 8, což mi poskytuje celkové rozlišení 1600 kroků na otočku. Toto rozlišení je dostatečné i pro pomalé pohyby, kdy je pohyb plynulý a nedochází k žádnému cukání. Nevýhodou vyššího rozlišení (většího mikrokrokování) je fakt, že je potřeba pro stejnou rychlost motoru generovat více pulzů a tím pádem jsou kladeny větší nároky na PC na generování těchto pulzů. Tento faktor začíná být velmi omezující při generování pulzů pomocí PC a ne např. pomocí HW interpolátoru, jelikož je potřeba mít velmi nízkou hodnotu odezvy z Latency testu, aby nedocházelo ke ztrátě kroku. Pokud tento případ nastane, je vhodné najít rozumný kompromis mezi velikostí mikrokrokování a maximální požadovanou rychlostí pojezdu a případně se pokusit snížit Latency systému. V mém případě je velikost mikrokrokování 8 zcela vyhovující, jelikož i při maximální rychlosti je nejvyšší frekvence generování pulzů cca. $\frac{1}{2}$ maxima, co zvládne počítač.

- Převod (poměr) mezi motorem a kuličkovým šroubem

V mém projektu je motor ke šroubu připojen napřímo, takže jedna otočka motoru znamená jednu otočku šroubu a tak jsem poměr zachoval 1:1.

- Stoupání kuličkového šroubu

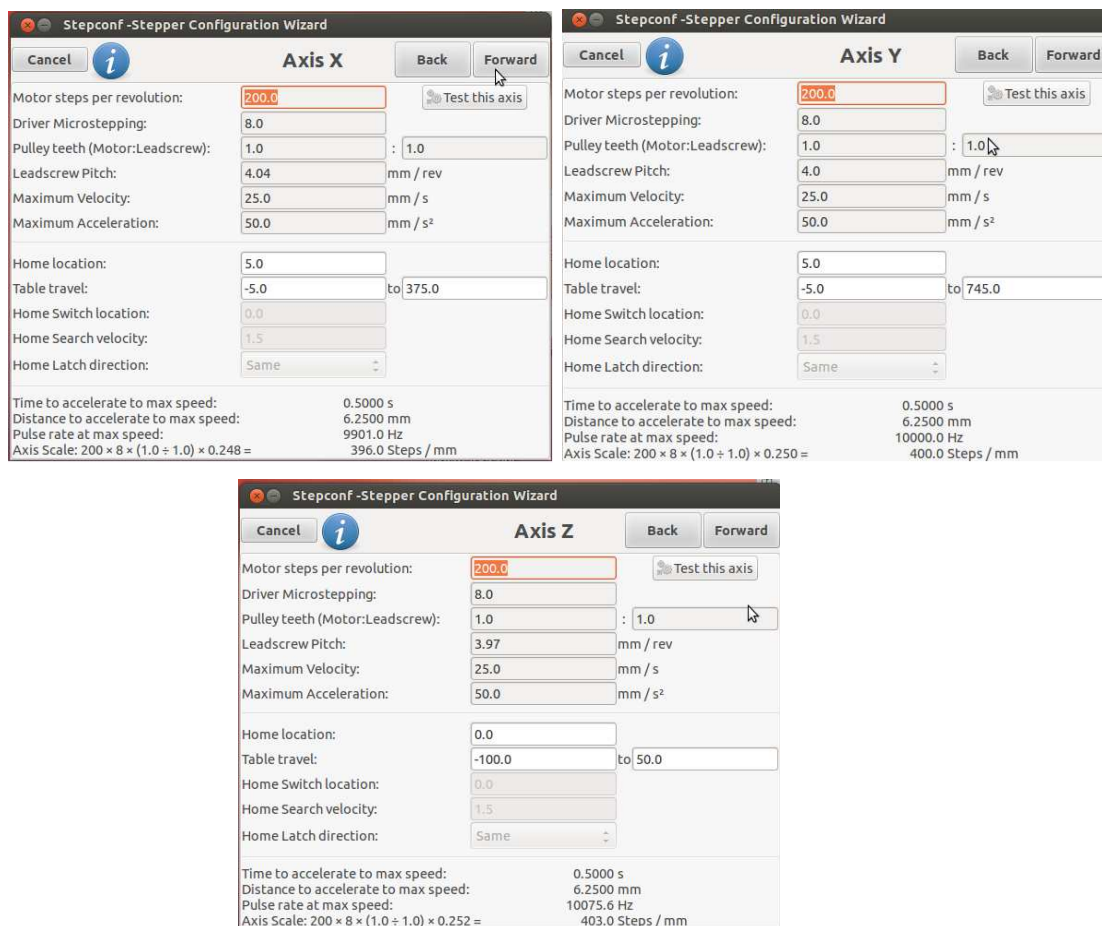
V mém případě 5mm/otočka. Toto nastavení jsem později upravil pro každou osu individuálně a to z důvodu nepřesnosti při polohování jednotlivých os. Odůvodnění přenastavení je uvedeno níže v textu viz. kapitola 6.

- Maximální rychlost a maximální zrychlení

Zvolil jsem maximální rychlost i zrychlení pro všechny osy stejné a to: max. rychlost: 25 mm/s , max. zrychlení: 50 mm/s².

- Domácí pozice frézy a rozsah pojezdů pro jednotlivé osy

Okna pro nastavení parametrů jednotlivých os obsahují navíc testovací tlačítko, díky kterému je možné nastavení okamžitě otestovat v praxi.



Obrázek 5.4: LinuxCNC - nastavení parametrů pro jednotlivé osy

Na začátku projektu jsem měl za cíl dosáhnout vyšších rychlostí a motory i elektroniku jsem na tyto rychlosti dimenzoval, avšak po kompletním zapojení a odzkoušení přímo na stroji jsem zjistil, že mechanická konstrukce stroje neumožňuje tak rychlý pohyb, jelikož dochází k rozvibrování portálu, hlavně při pohybu po nejdelší ose Y. Na základně toho jsem tedy upustil od tohoto požadavku a postupným zkoušením přímo na stroji našel optimální rychlost posuvu. Nastavená maximální rychlost je větší, než rychlost, kterou se strojem reálně pohybují ať už bez zátěže, či při obrábění a je to z toho důvodu, že plánuji konstrukci vyztužit a rychlost zvýšit.

Pokud ovládáme vřeteno pomocí PC, nastavujeme jeho parametry v okně „Spindle“. Jelikož vřeteno zatím ovládám manuálně, tomuto nastavení jsem se vyhnul.

V posledním okně stepconf wizardu („Options“) je možné do prostředí přidat možnost použití virtuálních kontrolních panelů. V závislosti na GUI, které volíme na samotném začátku při založení nové konfigurace, je možné přidávat do prostředí LinuxuCNC vlastní grafické kontrolní panely. Tato možnost není přítomná u všech GUI, jelikož pokud např. používáme „touchy“ interface, tak zde není možné přidávat vlastní grafické panely. Jelikož používám defaultní prostředí

„axis“, tuto možnost jsem měl. Při zaškrtnutí možnosti „Include custom PyVCP GUI panel“ je následně možné vytvářet si vlastní ovládací a zobrazovací prvky v prostředí LinuxCNC, jako jsou například posuvníky, ukazatele, indikační diody atd. Tyto prvky je možno propojit s různými fyzickými kontakty a přidat si tak možnost zobrazení, nebo alternace jakékoliv proměnné, které chceme, pokud nám tato možnost v defaultním nastavení chybí. V mém projektu jsem si tuto možnost ponechal, ačkoliv jsem doposud neměl potřebu si jakékoliv ovládací či zobrazovací prvky přidávat, jelikož defaultní nastavení rozhraní „axis“ obsahuje vše, co jsem doposud potřeboval.

Závěrem konfigurace stepconf wizard uloží všechna nastavení, co jsme provedli a vytvoří na námi zvoleném místě novou složku se všemi konfiguračními soubory. Jedná se hlavně o soubory *.hal a *.ini, ve kterých jsou uloženy veškeré konfigurační údaje tak, jak jsme je zadali do stepconf wizardu. Zároveň dojde k vytvoření spustitelného odkazu, kterým spouštíme přímo tuto konkrétní konfiguraci LinuxCNC.

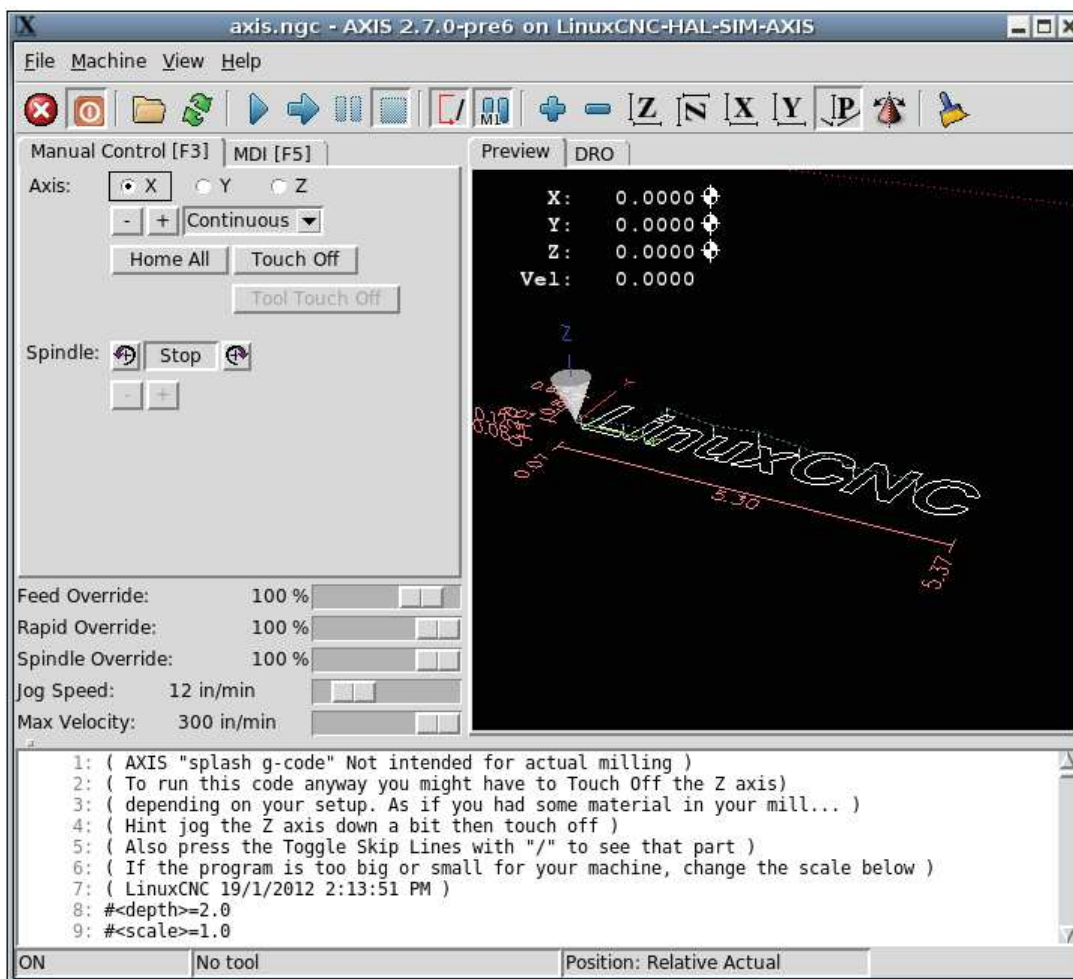
V případě potřeby úpravy konfigurace, je možné znovu spustit stepconf wizard, kde na začátku zvolíme možnost upravit již vytvořenou konfiguraci, projdeme znovu celým procesem a přepíšeme požadované hodnoty, nebo je možné konkrétní údaje přepsat přímo v konfiguračních souborech, a tím si činnost mírně urychlit.

5.3.3 Uživatelské prostředí – axis GUI

Při prvotním nastavování LinuxCNC máme na výběr z několika GUI, které lze různým způsobem upravovat. Při volbě prostředí je hlavním faktorem způsob, jakým budeme LinuxCNC ovládat (pomocí klávesnice a myši nebo dotykovou obrazovkou) a jak moc chceme systém upravovat. Dále můžeme samozřejmě grafické prostředí volit dle osobních preferencí.

Defaultní a nejčastěji používané GUI LinuxCNC je „axis“, které jsem zvolil i já pro svůj projekt, jelikož mi přišlo nejintuitivnější. Systém mi běží na PC a ovládám ho pomocí klávesnice a myši a zároveň prostředí „axis“ nabízí již zmíněnou možnost přidání dodatečných ovládacích panelů pomocí PyVCP. Dále tedy budu popisovat grafické prostředí „axis“ a jeho hlavní ovládací prvky.

Po správně provedeném nastavení všech parametrů nám stepconf wizard vytvoří mimo inicializačních a konfiguračních souborů navíc i spustitelný odkaz. Tento odkaz spouští přímo LinuxCNC s nastavením, které jsme dříve provedli. Při spuštění LinuxCNC s nastaveným prostředím „axis“ se nám spustí nejprve úvodní obrazovka, podobná následující.



Obrázek 5.5: LinuxCNC - axis GUI [22]

Nebudu zde popisovat význam každé jednotlivé ikony, což nakonec ani není předmětem práce. Popíšu hlavní ovládací prvky, se kterými jsem nejčastěji pracoval a následně postup, jak může ovládání frézky pomocí LinuxCNC vypadat. U horního okraje obrazovky, těsně pod rozbalovacími roletami se nachází pás tlačítek. První tlačítko resetuje E-STOP a druhým „oživujeme“ systém. Následují dvě tlačítka pro práci se soubory a sada čtyř tlačítek, pomocí kterých spouštíme, případně pozastavujeme vykonávání programu, tedy instrukcí, zapsaných v právě otevřeném souboru s G-kódem. Poslední sada ikon slouží k manipulaci s obrazem náhledu.

Největší část obrazovky zabírá náhled na nahraný soubor s G-kódem. Jedná se o cesty nástroje pro daný obrobek, přičemž při spuštění LinuxCNC se vždy nejdřív načte inicializační soubor, který obsahuje instrukce pro gravírování nápisu „LinuxCNC“. Vlevo od náhledu jsou viditelné dvě záložky („Manual control“ [F3] a „MDI“ [F5]).

Záložka „Manual control“ slouží k manuálnímu pohybu vždy jednou zvolenou osou pomocí tlačítek „+“ a „-“, případně i více osami zároveň, pomocí klávesových

zkratek. V bloku „Manual control“ dále můžeme tlačítkem home nastavit domácí pozici pro jednotlivé osy, případně tlačítka Spindle ovládat vřeteno, pokud tuto možnost podporuje HW (není můj případ). Pokud nastavíme u některé osy aktuální pozici jako její home, pak se tato změna projeví symbolem vedle dané osy v okně náhledu.

V záložce „MDI“ můžeme manuálně psát instrukce v G-kódu a ty následně nechat stroj vykonat. Pod těmito dvěma záložkami se nachází ještě pás s posuvníky, kde můžeme nastavovat další parametry, zejména pak maximální rychlost a zrychlení pro manuální pohyb. Pod posuvníky a náhledem, na spodní straně obrazovky se nachází pouze část výpisu z aktuálně otevřeného souboru s G-kódem.

Postup práce s LinuxCNC – tedy otevření a spuštění vygenerovaného G-kódu může v mém případě vypadat následovně:

- Spuštění LinuxCNC s platným nastavením
- Resetování E-STOP tlačítka [F1] a oživení systému [F2]
- Načtení souboru s G-kódem „File -> Open -> otevření souboru *.ngc“ a vizuální ověření správnosti cest na základě zobrazovaného náhledu
- Nastavení home pozice pro všechny osy – manuálně najedu frézou na pozici, kdy se dotýká horní stěny obrobku a to zhruba do bodu, který jsem při generování G-kódu nastavil jako počátek (zde se jedná o osobní preferenci, kde nastavíme počátek. Při práci s frézou jsem si zvykl nastavovat počátek do levého spodního rohu obrobku, na dotyk s jeho horní stěnou). Poté stiskem tlačítka „Home axis“ nastavím tuto pozici jako home pro všechny osy samostatně.
- Spuštění přívodu napětí do spínaného zdroje a do vřetena pomocí přepnutí kolébkového spínače na těle CNC frézky.
- Nastavení otáček a uvedení vřetene do chodu manuálním spínačem na jeho těle.
- Spuštění vykonávání instrukcí, zapsaných v načteném souboru pomocí tlačítka modré šipky v horní liště obrazovky.

5.4 CAD/CAM

Jak již bylo vysvětleno v rámci rešerše, CAD programy slouží k tvorbě 3D návrhu výsledného výrobku a následně za pomoci CAM programu dojde k převedení tvaru takového návrhu na cesty pro nástroj a následně na G-kód. V rámci rešerše jsem zmínil některé CAD/CAM programy, přičemž v provozu jsem otestoval dvě kompletní řešení, jejichž rozdíly a výběr vhodnějšího nyní popíšu. Záměrně nebudu popisovat vysloveně rozložení a práci s CAD programy, jelikož to není předmětem práce a jednalo by se o velmi zdlouhavý popis. Místo toho popíšu

hlavní rozdíly mezi oběma použitými CAD/CAM řešeními a důvody, proč jsem zvolil jako výsledné řešení kompletní balík od Autodesku – Fusion 360. Následně se budu věnovat nastavení CAM softwaru, jelikož v tomto případě je již potřeba nastavit konkrétní parametry pro obrábění.

Pro grafický návrh je v podstatě možné použít jakýkoliv CAD program a výsledek bude stejný. Rozdíly ovšem nastanou mezi jednotlivými CAM programy, jelikož každý z nich pro onen návrh generuje jinou cestu nástroje. Rozdíly v různých cestách mohou být velice markantní, stejně tak i rozdíly při samotném nastavování CAM softwaru. Výsledná volba CAM softwaru má tedy vliv nejen na efektivitu obrábění, ale jak jsem si nakonec potvrdil až v praxi, tak i na samotnou smysluplnost a správnost vygenerovaného G-kódu.

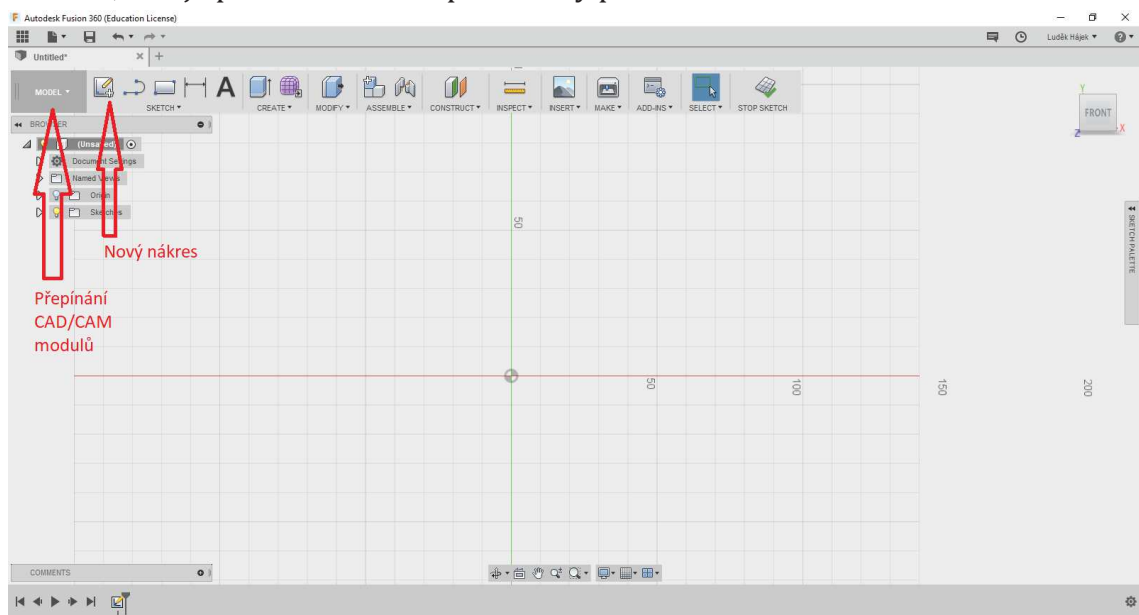
5.4.1 Google SketchUp + SketchUcam

Prvním programem, se kterým jsem pracoval, byl CAD program SketchUp. Jedná se o volně stažitelný SW, který je velice intuitivní a přitom obsahuje naprosto všechny nástroje a možnosti, které jsem kdy potřeboval. Další velkou výhodou, která je čistě osobní, je fakt, že s programem SketchUp jsem již dříve pracoval a znám jeho prostředí a ovládání.

Problém ovšem nastal, když jsem z grafického návrhu chtěl vygenerovat G-kód. Do SketchUpu se dá nainstalovat dodatečný plugin SketchUcam. Pomocí tohoto pluginu jsem tedy byl schopný vygenerovat G-kód na základě dříve vytvořeného grafického návrhu. Již samotné nastavování parametrů v rámci tohoto pluginu ovšem bylo velmi neintuitivní. Hlavní problém však nastal až ve fázi, kdy jsem výsledný soubor s G-kódem vygeneroval a chtěl jej importovat do LinuxCNC. V první řadě se mi nepovedlo v nastavení exportu zvolit takový formát souboru, který je možné v LinuxCNC otevřít. Zároveň při revizi vygenerovaného G-kódu a jeho porovnání s G-kódem vygenerovaným programem Fusion 360 jsem zjistil, že cesty, které vygeneroval SketchUcam naprosto neodpovídají realitě obrobku. Mám tím na mysli skutečnost, že i pokud bych stroj nechal, aby se řídil tímto konkrétním G-kódem, výsledek by naprosto neodpovídal zadání. O rozdílnosti různých cest ke stejnému výsledku jsem věděl, avšak zde vygenerovaný výsledek zcela neodpovídal zadání. V neposlední řadě možnosti nastavení SketchUcam byly značně omezené a osobně mi nastavování přišlo nepřehledné. SketchUp i Fusion 360 jsem testoval prakticky souběžně a jelikož CAM část Fusionu 360 mi osobně přišla mnohem přehlednější a zároveň, až na malé drobnosti, které popíšu později, byl i vygenerovaný G-kód zcela správný, rozhodl jsem se řešení od SketchUpu opustit a dále pracovat již jen s Fusionem 360.

5.4.2 Autodesk Fusion 360

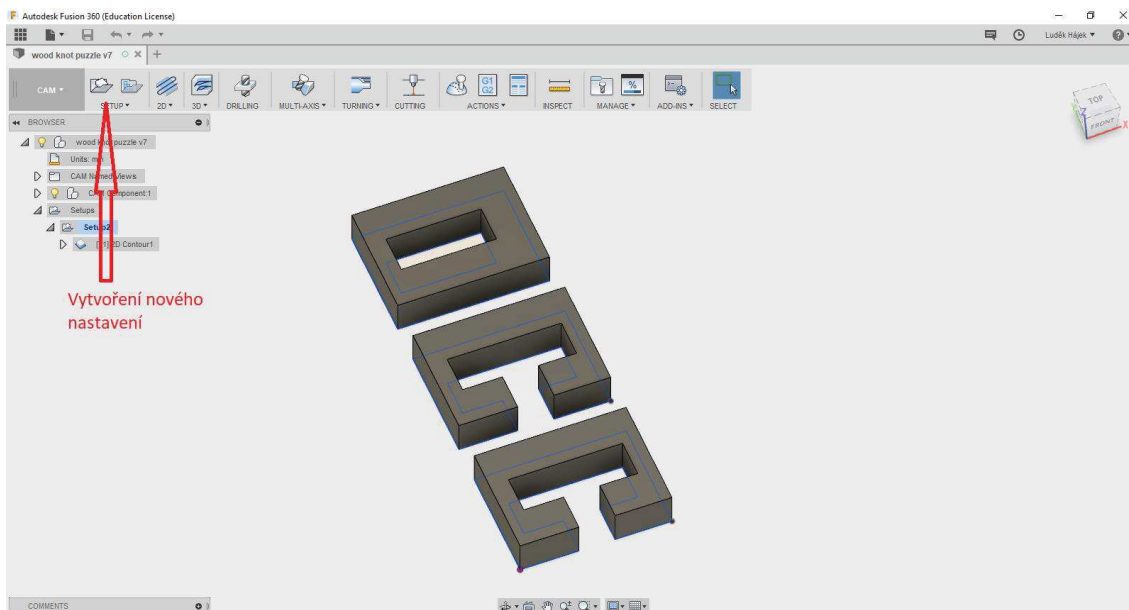
Autodesk Fusion 360 poskytuje vše potřebné pro úspěšné vytvoření návrhu a následné vygenerování G-kódu. Zároveň je možné stáhnout studentskou verzí zdarma. Mezi CAD a CAM částmi programu se velmi jednoduše přepíná pomocí panelu v levém horním rohu. Při spuštění se automaticky načte CAD modul s názvem „MODEL“. Jedná se o klasický 2D/3D CAD, který slouží ke grafickému návrhu obrobku. Vzhledem k povaze a zadání mé bakalářské práce nebudu CAD část programu Fusion 360 popisovat, jelikož je velice podobná jakémukoliv jinému CAD programu. Postup práce je takový, že první tlačítko vedle přepínače modulů v levém horním rohu slouží k vytvoření nového návrhu. Po jeho stisku vybereme v souřadnicových osách tu, na kterou chceme nakreslit první náskres. Můžeme vytvořit několik náskresů a ty potom pomocí funkcí totožných s ostatními 3D CAD programy převádět na 3D modely. Jakmile máme grafický návrh obrobku hotový, tlačítkem „STOP SKETCH“ náskres ukončíme a můžeme se přepnout do CAM modulu, kde je potřeba nastavit parametry pro frézování.



Obrázek 5.6: Fusion 360 - CAD

5.4.2.1 Nastavení CAM

Po přepnutí do CAM modulu dojde k několika změnám. Nejvýraznější z nich je změna ovládacích prvků v horní části obrazovky. Namísto tlačítek pro tvorbu a editaci náskresu se nyní v tomto panelu nachází tlačítka pro nastavení obrábění, simulace, atd. Hlavní část obrazovky se ovšem nemění, jelikož stále vidíme grafický návrh.



Obrázek 5.7: Fusion 360 - CAM

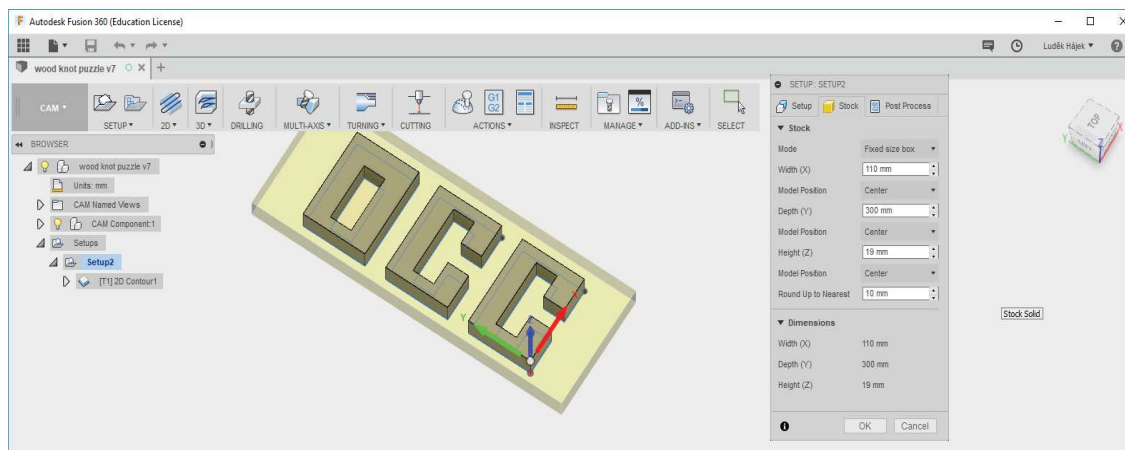
Postup při nastavování CAM může vypadat následovně:

- Vytvoření nového nastavení
- Volba a nastavení jedné či více frézovacích operací z karet 2D/3D
- Spuštění simulace – vygenerování cest nástroje a kontrola, jestli nedochází ke kolizím a neexistují neodfrézovaná místa
- Vygenerování G-kódu

Prvním krokem při nastavování CAM programu je vytvoření nového nastavení. V horním panelu tedy vybereme „Setup -> New setup“. Objeví se okno, kde zadáme několik základních nastavení.

Nastavujeme:

- Typ operace (frézování/ soustružení/ řezání)
- Souřadnicový systém pro práci a bod počátku – zde se jedná o individuální nastavení dle osobních preferencí, kde chceme, aby vůči obrobku ležel počátek.
- V kartě „Stock“ nastavíme parametry obrobku, čili surového materiálu, ze kterého chceme obrábět. Lze nastavit buď relativní velikost, kdy Fusion 360 sám vytvoří materiál okolo našeho návrhu a můžeme nastavit s jakým přesahem, nebo fixní velikost, kdy musíme zadat přímo rozměry obráběného materiálu.
- V tuto chvíli stačí kliknout na „OK“ a vytvoří se počáteční konfigurace.



Obrázek 5.8: Fusion 360 - CAM - New setup

S takto vytvořenou konfigurací již můžeme přejít k samotnému nastavení obrábění. V závislosti na požadovaném způsobu obrábění volíme jednu či více obráběcích operací. Fusion 360 nabízí velké množství takovýchto operací. Hlavní nastavení pro frézování jsou přístupná pod dvěma rozbalovacími ikonami (2D a 3D), přičemž 2D obsahuje i taková nastavení, kde se reálně pohybujeme ve třech osách.

V kategorii 2D jsou takové operace (nastavení frézování), kdy než se fréza začne pohybovat v osách X a Y, nastaví si nejprve osu Z na konstantní výšku, kterou následně neměníme. Jakmile dojde k odfrézování jedné vrstvy dle zadané kontury, fréza se v ose Z sníží, dle nastaveného úběru a opět můžeme odfrézovat jeden průchod. Výsledkem jsou poté vyfrézované plochy, které jsou vždy rovnoběžné s podstavou, ať už se jedná o drážky, nebo větší plochu.

3D nastavení již umožňují i taková frézování, kdy se mění výška v rámci frézované plochy. Frézovat se tak dají jakékoliv rampy, zakřivené a sférické plochy, atd.. Nyní popíšu nastavení pro základní způsoby obrábění, taková, se kterými jsem přímo pracoval nebo pracovat budu na dalších projektech.

Základní operace, které lze ve Fusionu 360 nastavit:

- 2D Adaptive clearing a 2D Pocket – rychlé vyfrézování materiálu z velkých ploch – zvolené 2D kontury jsou dodržovány s větší tolerancí a důraz je kladen na plynulost operace
- 2D Face – použití velké frézy pro odstranění materiálu z povrchu – začistění horní frézované plochy
- 2D Contour – sleduje zadanou 2D konturu do zadané hloubky – např. vyfrézování vnějšího tvaru
- 2D Slot – frézování drážky podél zadané osy

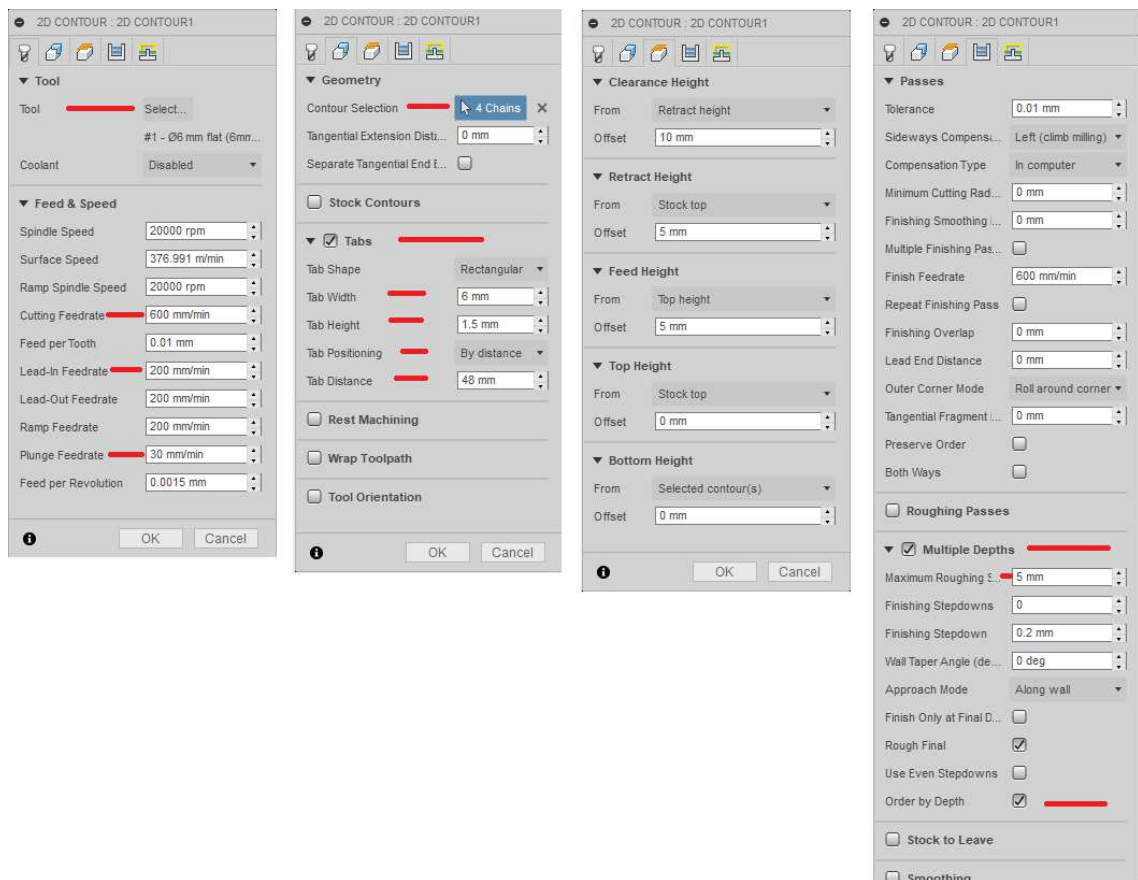
- 3D Adaptive clearing a 3D Pocket clearing – Frézování zadané plochy u které se může spolu s osami X a Y souběžně měnit i osa Z. Opět se jedná o hrubou frézovací strategii, kdy je následně pro přesný výsledek potřeba dočištění. Slouží k rychlému a efektivnímu odstranění velkého množství materiálu.
- 3D Parallel – Vytvoří dráhy nástroje, paralelní k rovině XY. Tento nástroj je vhodný, pokud nechceme frézovat příliš strmé plochy.
- 3D Contour – Nastavení, vhodné pro frézování strmých stěn.
- 3D Scallop – Kompromis mezi 3D Parallel a 3D Contour. Frézování probíhá od vnějších okrajů dovnitř obrobku a snaží se držet konstantní vzdálenost mezi jednotlivými průchody. Toto nastavení se často používá jako dokončovací strategie po předešlých hrubých operacích, které zanechávají přebytečný materiál.

Jak je možné vidět z výčtu operací, každá vytvoří pro stejně zadanou plochu rozdílné cesty nástroje dle toho, k čemu je určena. V praxi to tedy znamená, že pokud potřebujeme odfrézovat složitější zakřivené plochy, je potřeba kombinovat několik operací k dosažení optimálního výsledku. Volbu konkrétní operace provádíme na základě toho, k čemu je určena a dle zkušenosti, kdy podle frézované plochy víme, jaká operace poskytne nejlepší výsledek. Důležité ovšem je, vždy si zkontrolovat v rámci simulace vygenerované cesty nástroje a zkontrolovat, obzvláště u složitějších ploch, že se na obrobku nenacházejí místa, kde by docházelo k nedostatečnému pokrytí nástrojovými drahami. V tom případě by zůstával neodfrézovaný materiál a výsledek by neodpovídal zadání. Pro každou operaci můžeme dále nastavit jinou frézu a výsledek je potom kombinací 2D a 3D operací s různými nástroji.

Dejme tomu, že chceme vyrobit malou misku. Příklad použití různých operací v rámci frézování jednoho obrobku může vypadat následovně. Nejprve pomocí 2D Face, za použití velké frézy, srovnáme horní plochu do ideální hladké roviny. Následně nahrubo vyfrézujeme vnitřní objem, čímž vytvoříme prohloubení misky. Na tuto hrubou operaci můžeme použít např. 3D Adaptive clearing, případně 3D Contour, dle strmosti stěn misky. V dalším kroku pomocí 3D Scallop a nastavení menšího přesahu odfrézujeme zbylý materiál z vnitřku misky a docílíme tím hladkého povrchu. V posledním kroku oddělíme misku od zbylého materiálu za pomocí 2D Contour, kdy budeme sledovat vnější obvodovou hranu misky.

Pro každou zvolenou operaci je třeba nastavit několik základních parametrů. Nastavované parametry se mohou lehce lišit v závislosti na typu operace. Jakmile zvolíme požadovanou operaci, otevře se nám okno s jejím nastavením. Jako příklad uvedu, jaké parametry je potřeba zadat při nastavování 2D Contour. Záměrně

uvádím 2D Contour, jelikož se jedná o jednu z nejčastěji používaných operací pro jednoduché vyřezání dílů a použil jsem ji i při frézování svého prvního výrobku.



Obrázek 5.9: Fusion 360 - nastavení parametrů 2D Contour

V první záložce „Tool“ vybereme pomocí tlačítka „Select“ z nabídky nástrojů takový, jaký budeme používat. Dále nastavujeme rychlosti pro frézování. První tři boxy obsahují nastavení pro vřeteno, a jelikož mám vřeteno řízené manuálně, můžu klidně nastavené hodnoty ponechat, případně nastavit otáčky vřetena „Spindle speed“ na hodnotu, kterou následně nastavím manuálně. Důležité jsou ovšem následující hodnoty:

- „Cutting feedrate“ - rychlost posuvu os při frézování (jak maximálně rychle bude fréza procházet materiálem) [mm/min].
- „Lean-in feedrate“ a „Lean-out feedrate“ – rychlosti pohybu, když fréza vjíždí do a vyjíždí z materiálu. [mm/min]
- „Plunge feedrate“ – rychlost, kterou fréza zajíždí do materiálu při vertikálním pohybu, tedy když sestupuje o jedno patro níž.

Druhá záložka „Geometry“ obsahuje dvě hlavní nastavení. Prvním je volba kontur „Contour selection“, které chceme sledovat. Zde je potřeba si uvědomit, že

při nastavování parametrů musíme vždy definovat přesně to, co chceme, aby stroj udělal. Pokud tedy chceme například vyřezat nějaký tvar z dřevěné desky a máme udělaný 3D návrh o tloušťce této desky, je potřeba ohraničující konturu zvolit na spodní straně, aby došlo ke kompletnímu prořezání. Druhým nastavením jsou „Tabs“. Zde nastavujeme přichytné body. Jedná se o malé spojovací body mezi zbylým materiálem a výrobkem. Fréza při posledním průchodu podél kontury neprořeže celou dráhu, ale na předem určených místech ponechá materiál, který bude výrobek spojovat se zbylou masou. Je to z toho důvodu, aby se při posledním průchodu výrobek neodtrhnul a nezničil sebe i frézu. Rozměry těchto přichytných bodů lze nastavit, stejně jako jejich umístění. Můžeme je rozmístit buď po určité zadané vzdálenosti, nebo tato místa vyznačíme manuálně přímo na vybraných konturách.

Ve třetí záložce „Heights“ nastavujeme výšky. Nastavíme výšku spodní a horní stěny obrobku a také výšky, ve kterých má fréza jezdit při přejezdu naprázdno. Můžeme si vybrat, vůči čemu výšky definujeme a případně přidat offset.

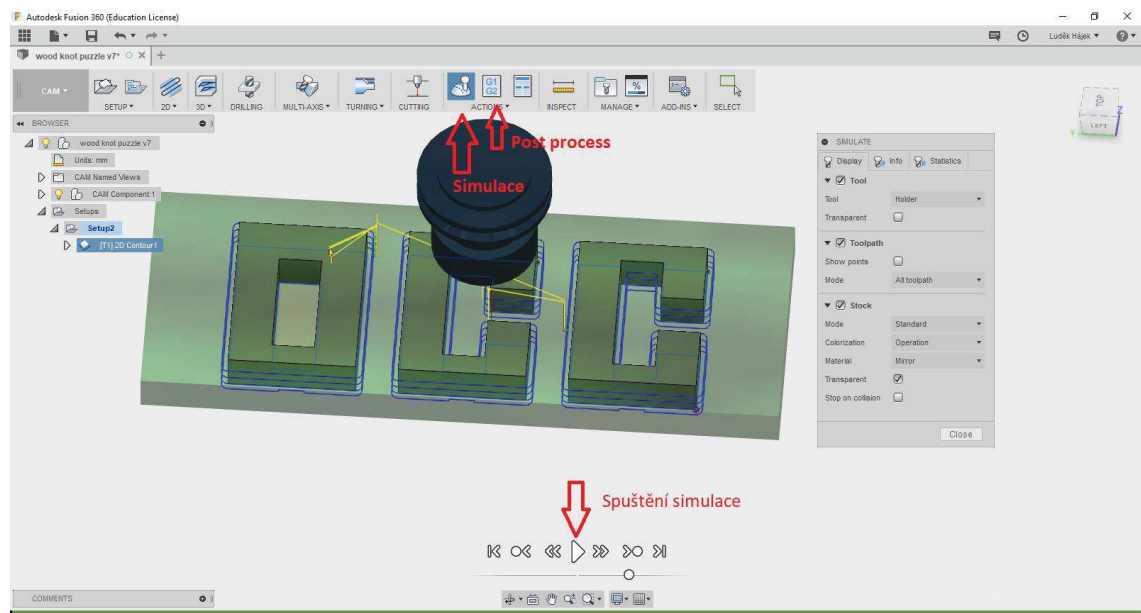
Čtvrtá a poslední záložka, která obsahuje nezbytná nastavení, je záložka „Passes“. Zde můžeme nastavit toleranci, stejně jako nastavení pro různé průchody, obzvláště potom, pokud chceme při posledním průchodu snížit tolerance a docílit tak co nejlepšího povrchu a dodržení zadaných kontur. Pro běžné aplikace ovšem můžeme tato nastavení nechat defaultní, musíme však nastavit maximální hloubku frézování při jednom průchodu. Zaškrtnutím možnosti „Multiple passes“ se nám zobrazí blok políček. Zde zadáme maximální hloubku pro jeden průchod „Maximum roughing stepdown“ a zároveň, pokud máme označeno více kontur, jestli má frézování probíhat v pořadí dle oblasti, nebo dle hloubky průchodu.

Jakmile máme všechny parametry nastavené, stiskem tlačítka OK nastavení potvrdíme a Fusion 360 vygeneruje nástrojové cesty pro tuto operaci. Obdobným způsobem nastavíme více operací, je-li to potřeba.

5.4.2.2 Simulace a generování G-kódu

Posledním krokem před vygenerováním G-kódu je kontrola cest nástroje pomocí simulace. Stiskem tlačítka „simulate“ v horní liště spustíme simulaci. Náhled na grafický návrh se změní a objeví se okno s parametry pro simulaci. Zde si můžeme zvolit, zda chceme v simulaci vidět cesty nástroje, frézovaný materiál, nebo frézu. Zároveň se v záložce „Statistics“ objeví předpokládaný čas obrábění. Na spodní straně obrazovky pod grafickým návrhem přibyla časová osa spolu s ovládacími prvky simulace. Tlačítkem „Start the simulation“ simulaci spustíme a vizuálně kontrolujeme, zda nástroj provádí pohyby, které má. Pokud by mělo dojít ke kolizi frézy např. s podložkou, na zelené časové ose se objeví červené úseky a popisem,

co se přesně stalo. Pokud simulace proběhne úspěšně a osa zůstane celá zelená, nastavení je provedeno správně a můžeme z cest nástroje vygenerovat G-kód.

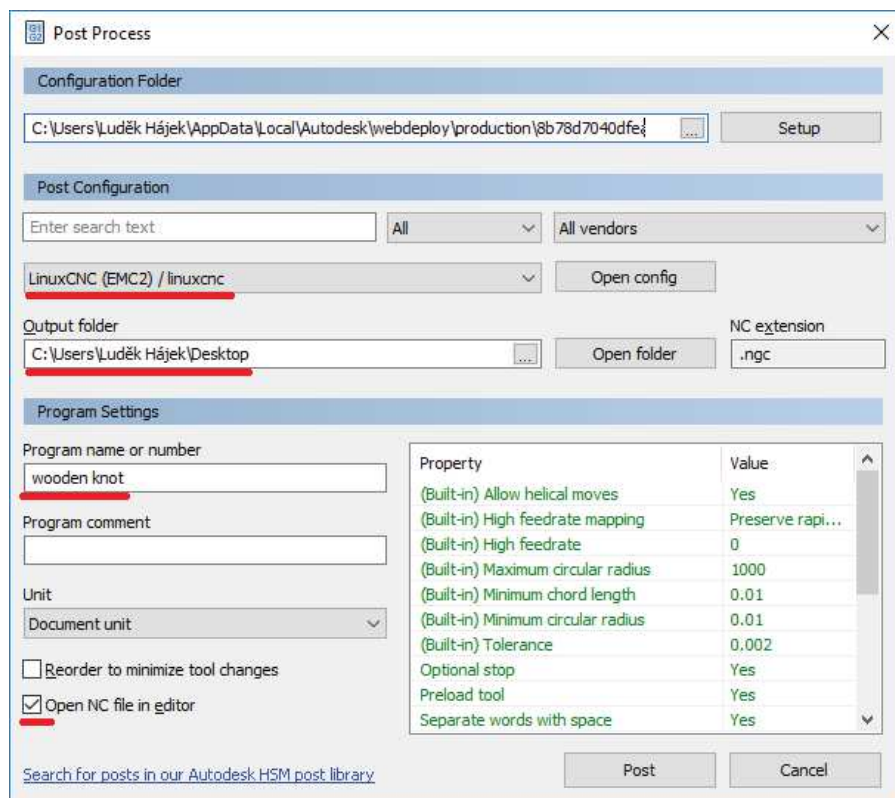


Obrázek 5.10: Fusion 360 - Simulace

Stiskem tlačítka „Post process“ otevřeme nastavení post processingu. Jedná se o proces převodu strojově nezávislých údajů o poloze frézy na strojově závislý G-kód. Výstupem je tedy G-kód, který bez dalších úprav lze načíst a spustit přímo v LinuxCNC.

Nastavení post processingu:

- Z rozbalovací nabídky vybereme LinuxCNC (EMC2) – automaticky se nám upraví typ souboru na *.ngc.
- Nastavíme výstupní složku pro vygenerovaný G-kód (Output folder)
- Pojmenujeme výstupní soubor
- Ponecháme označenou možnost pro otevření NC kódu v editoru, jelikož ještě bude potřeba provést jeho drobné úpravy
- Tlačítkem „Post“ vygenerujeme G-kód



Obrázek 5.11: Fusion 360 - nastavení Post processingu

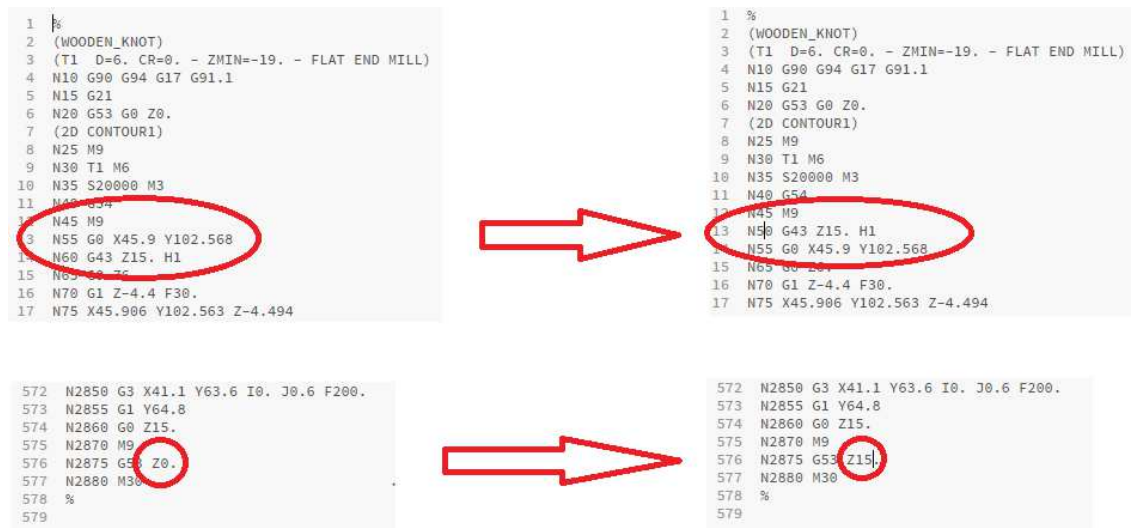
5.4.2.3 Úprava vygenerovaného G-kódu

Po vygenerování se soubor s instrukcemi automaticky otevře. Nyní již máme náhled na samotný G-kód, podle kterého se, po jeho nahrání do programu LinuxCNC, budou řídit pohyby jednotlivých motorů. Vygenerovaný G-kód je ovšem ještě potřeba lehce upravit. Jedná se o změnu výškových parametrů. Zatím nevím proč, ale kód vygenerovaný Fusionem 360 je celý správně, až na dva úseky - nastavení osy Z na začátku programu a na jeho konci.

Na začátku kódu, Fusion 360 vygeneruje nejprve instrukce pro přesun v osách X a Y z domácí pozice na první pozici na obrobku, kde bude frézovat. Následují řádky s instrukcemi pro posun pouze v ose Z na nastavenou výšku pro přejezd na volno a ihned za tím změna výšky na hladinu, ve které budeme frézovat při prvním průjezdu. Toto nastavení ovšem nedává smysl, jelikož pokud nastavím domácí pozici tak, aby se fréza dotýkala horní stěny obrobku, pak při přesunu např. doprostřed, odkud chci začít frézovat, se může fréza zlomit. Je tedy nutné, před řádek obsahující instrukce pro pohyb v rovině XY, přesunout instrukci nastavující výškovou hladinu pro přejezd naprázdno. Tím docílíme správného postupu, tedy že nejprve se osa Z zvedne do požadované výšky, následně s vřetenem najedeme nad počáteční bod frézování na obrobku a následuje spuštění vřetene a začátek samotného frézování.

Druhým místem, kde je kód potřeba přepsat je samotný konec programu. Poslední instrukce pro pohyb nastaví hodnotu osy Z na námi zvolenou polohu pro volný přesun a následuje instrukce pro opětovné spuštění do výšky, ve které jsme pro osu Z nastavili domácí pozici. Tím pádem při dokončení frézování dojde k opětovnému spuštění vřetene na úroveň, kde se fréza dotýká již hotového výrobku. Díky tomu, že vřeteno musím vypínat manuálně, může dojít ke znehodnocení výrobku. Je tedy vhodné v této poslední instrukci přenastavit hodnotu pro osu Z na takovou výšku, v jaké chceme, aby vřeteno zůstalo po dokončení frézování. Díky tomu nedojde jednak k poškození výrobku a zároveň budeme mít usnadněnou jeho demontáž.

Při spuštění takto upraveného G-kódu dojde k vyfrézování výrobku přesně dle zadání. Nevím, proč CAM generuje kód s těmito chybami. Je možné, že nutnost úprav je způsobena špatným nastavením CAM programu. Chyby jsem odhalil až na základě testování, kdy mi, jak jsem již psal, fréza nejprve najela na počáteční bod frézování, pak se zvedla a zase spustila. Otevřel jsem si tedy soubor s G-kódem a řádek po řádku ho začal revidovat. Zjistil jsem, že vygenerované pořadí řádků nedává logicky smysl. Po manuální úpravě již program funguje tak, jak má. Situaci řeším manuální editací vygenerovaného kódu, jelikož se jedná o jednoduchý proces, který je pokaždé stejný. Chyba se projevuje vždy na stejných řádcích, takže manuální oprava je pro mě v tuhle chvíli přijatelné řešení.



Obrázek 5.12: Editace vygenerovaného G-kódu

6 ZKUŠEBNÍ FRÉZOVÁNÍ A PLÁNY DO BUDOUCNA

Při testování stroje v praxi jsem zjistil, že konstrukce stroje je vůči zbylé elektronice poddimenzovaná. Při pohybu po nejdelší ose Y dochází ve větších rychlostech (nad 800 mm/min) k rozechvění portálu. Tento jev přetrvává i při zátěži, kdy se chvění promítne do kvality výsledného výrobku.

Rozechvění osy Y lze předejít snížením rychlosti lineárního posuvu na cca 500 mm/min. Tato rychlost se zdá být po odzkoušení prozatím maximální, při které nedochází ke znatelnému chvění a kvalita opracování výrobku je dostačující.

CNC frézka je konstruována primárně pro hobby opracovávání dřeva, tudíž nároky na přesnost a kvalitu výrobku jsou relativně malé. Při testovací výrobě dřevěného hlavolamu s finální konfigurací a následné kontrole výrobku jsem zjistil, že největší naměřená nepřesnost lineárního posuvu byla cca 0,8 mm na hraně dlouhé 80 mm. Odchylka je z největší části způsobena nepřesným nastavením parametrů osy Y v konfiguraci LinuxCNC. Po prvotním zapojení a odzkoušení přesnosti posunů jednotlivých os jsem zjistil, že vzdálenost, kterou fréza reálně urazila se hrubo neshodovala se zadanými parametry. Nastavení pro všechny osy byla provedena dle teoretických hodnot daných výrobcí, avšak při zadání lineárního posuvu jednotlivými osami o vzdálenost 100 mm, docházelo k nepřesnostem 18 mm na ose X, 27 mm na ose Y a 9 mm na ose Z. Provedl jsem tedy nejjednodušší korekci, a to přepočítání a úpravu nastavené hodnoty stoupání kuličkového šroubu. Po sérii přepočtů a odzkoušení má nyní stroj odchylku maximálně 1 mm na dráze dlouhé 100 mm. Tato odchylka ještě může být zmenšena pomocí přesnější úpravy hodnoty stoupání kuličkového šroubu, avšak pro tuto chvíli jsem s výsledkem spokojený a prioritou se pro mě stal problém s chvěním portálu.

Plány do budoucna:

- Vyztužení konstrukce portálu
- Dopojení E-STOP tlačítka
- Upgrade vřetene a jeho propojení s LinuxCNC
- Nalakování konstrukce
- Eliminace nepřesnosti lineárního posuvu

V současné chvíli je CNC frézka plně provozuschopná a vyrobil jsem na ní už několik testovacích výrobků. Nejdůležitějším bodem pro vylepšení se pro mě nyní stal problém s chvěním portálu, jelikož je tím omezen celkový potenciál stroje.

Řídicí elektronika, stejně jako rozměry stroje a výkon vřetena jsou koncipovány na větší rychlosti a tím pádem se malá rychlost posunu stává limitujícím faktorem pro použití, obzvláště při frézování komplexnějších výrobků. Frézka je ovšem i tak plně funkční a pro jednodušší operace, kde je výsledný čas krátký, není rozdíl zase tak markantní.

ZÁVĚR

Cílem mé bakalářské práce bylo sestrojít konstrukci a navrhnout řízení 3-osé CNC frézky pro hobby použití, s cílem opracovávat hlavně dřevo, případně po odzkoušení i hliník.

V prvotní části práce jsem se tedy zaměřil na návrh konstrukčního řešení stroje, dimenzování motorů a provedl jsem rešerši v oblasti řízení CNC strojů. Zvolil jsem konstrukční řešení s pohyblivým portálem. Jednotlivé osy jezdí po podepřených tyčích SBR20 o průměru 20 mm a o přenos krouticího pohybu motoru na lineární pohyb osy se starají kuličkové šrouby SFU/RM 1605 o průměru 16 mm. Celkové rozměry pojezdů jsou 1000 x 600 x 300 mm. Jako vřeteno jsem kvůli výhodné ceně použil přímou brusku od firmy DWT s příkonem 600W a možností regulace otáček 12000-27000 ot/min. Samotná konstrukce stroje, jednotlivé bočnice a nosné díly jsou vyrobeny z materiálu, který jsem měl dostupný na dílně. Bočnice stroje jsou ze 3mm silného, ztuženého plechu a konstrukce stolu a portálu z ocelových uzavřených tenkostěnných profilů (Jackel). S konstrukcí mi pomáhal otec strojař a veškeré konstrukční návrhy byly dělány formou náčrtů na papír, takže k této části neexistuje žádná reálná výkresová dokumentace.

Další částí práce bylo vybrat vhodné pohony jednotlivých os a pokusit se je správně dimenzovat. Zvolil jsem pohon krokovými motory Microcon SX 24-2341 s krouticím momentem 4,1 Nm. Motory jsou záměrně předdimenzované, jelikož přesný výpočet krouticího momentu motorů je velice obtížný a hlavně silně nepřesný, protože existuje mnoho parametrů, které na výsledek mají velký vliv a jsou neznámé, nebo s nimi nelze počítat. Tyto motory jsem nakonec zvolil proto, že k řízení nepotřebují zpětnou vazbu a mají dostatečný výkon na to, aby nedocházelo ke ztrátě kroku. Motory jsou zároveň schopné vyvinout rychlost až 1000 RPM.

Provedl jsem rešerši v oblasti řízení CNC strojů, abych zjistil, jaké všechny komponenty budu k řízení potřebovat a jak lze toto řízení realizovat. Rozhodl jsem se použít zapojení s PC -> breakout board -> drivery KM -> KM. Použité PC je jedno ze starých, vyřazených PC, co jsem měl doma. Na PC je nainstalován operační systém Ubuntu 12.04.5 LTS s RTAI jádrem a LinuxCNC. Breakoutboard jsem koupil spolu s drivery v sadě z Číny. Použité drivery jsou Leadshine M542, které umožňují napájet krokové motory do proudu až 4,2 A, napětí až 50V a zvolil jsem mikrokrokování 1/8. O přívod napětí na drivery se stará spínaný zdroj 800W 50V 16A, rovněž objednaný z Číny.

Po odzkoušení a porovnání jsem jako CAD/CAM program zvolil Autodesk Fusion 360. Mohu v něm navrhnout grafický návrh výrobku, ze kterého následně program dle nastavení frézování vytvoří cesty nástroje a vygeneruje G-kód. Ten po drobné úpravě nahraju na flash disk a přenesu do PC s LinuxCNC. Zde již kód mohu

v programu spustit a začít frézovat. Kompletní řetězec událostí od grafického návrhu po výrobek vypadá následovně: CAD -> CAM -> LinuxCNC -> Výrobek.

Při testovacím frézování jsem narazil na problém s tuhostí konstrukce portálu. Při vyšších rychlostech pohybů dochází k rozvibrování portálu a degradaci výsledného výrobku. Tento problém prozatím řeším snížením rychlosti na cca 500 mm/min. Při této rychlosti mám odzkoušené, že se portál nerozvibruje a výsledný výrobek odpovídá kvalitou i rozměry zadaným požadavkům. Tento problém plánuji v nejbližší době řešit vyztužením konstrukce portálu, jelikož se jedná o nejslabší článek celého stroje, který limitně omezuje potenciál frézky, zejména pak její elektroniky.

Literatura

- [1] Šablona pro BP/DP a prezentace v2.63 [online]. Brno: FEKT VUT, 2017 [cit. 2017-03-06]. Dostupné z: <http://latex.feec.vutbr.cz/sablona/>
- [2] CNC – Hobby: Stránky o CNC pro volný čas [online] [cit. 29. 10. 2017]. Dostupné z: <http://www.cnc-hobby.cz/>
- [3] Lineární vedení a ložiska. 4ISP [online]. [cit. 29. 10. 2017]. Dostupné z: <https://cnc.inshop.cz/linearni-vedeni-a-loziska/>
- [4] Posuvné šrouby a spojky. 4ISP [online]. [cit. 30. 10. 2017]. Dostupné z: <https://cnc.inshop.cz/posuvne-srouby-a-spojky/?pageindex=1>
- [5] Ozubené hřebeny. CNCshop.cz [online]. [cit. 30. 10. 2017]. Dostupné z: http://www.cncshop.cz/ozubene-hrebeny-kola_c
- [6] CNC frézka – mechanika [online]. [cit. 30. 10. 2017]. Dostupné z: <http://mujweb.cz/hmodel/pages/cnc.htm>
- [7] Stejnoseměrné stroje (MOTORY) [online]. [cit. 6. 11. 2017]. Dostupné z: <http://fei1.vsb.cz/kat430/old/Studium/Materialy/VS/kap22.pdf>
- [8] Electropaedia – „Electric Drives – Brushless DC/AC and Reluctance Motors“ [online]. [cit. 6. 11. 2017]. Dostupné z: <http://www.mpoweruk.com/motorsbrushless.htm>
- [9] Robotika.cz - Krokové motory [online]. [cit. 10. 11. 2017]. Dostupné z: <https://robotika.cz/articles/steppers/cs>
- [10] NOVÁK, Petr. Mobilní roboty: pohony, senzory, řízení. Praha: BEN - technická literatura, 2005. ISBN 80-7300-141-1.
- [11] Materiály ke zkoušce – předmět BPPA, VUT, FEKT. [online]. [cit. 10. 11. 2017]. Dostupné z: [intraportálu VUT po přihlášení na kartě předmětu „Prostředky průmyslové automatizace“, autor prezentace není uveden]: https://www.vutbr.cz/studis/student.phtml?gm=gm_detail_predmetu&apid=184049
- [12] HÁJEK, R. Konstrukční návrh os frézky, Brno, Vysoké učení technické v Brně, Fakulta strojního inženýrství. 2017, 66 s., Vedoucí diplomové práce Ing. Jan Vetiška, Ph.D.
- [13] Hybridní dvoufázové krokové motory řady SX. MICROCON. [online]. [cit. 10. 11. 2017]. Dostupné z: <http://www.microcon.cz/>
- [14] CNC router source – Breakout boards. [online]. [cit. 15. 11. 2017]. Dostupné z: <http://www.cncroutersource.com/breakout-boards.html>
- [15] CNC router source – Stepper motor control: motor drivers. [online]. [cit. 15. 11. 2017]. Dostupné z: <http://www.cncroutersource.com/stepper-motor-control.html>

- [16] SKALICKÝ, Jiří. Elektrické servopohony. Vyd. 2. Brno: Vysoké učení technické, 2001. ISBN 80-214-1978-4.
- [17] BORSKÝ, Václav. Základy stavby obráběcích strojů. Vyd. 2., přeprac. Brno: VUT, 1991. Učební texty vysokých škol (Vysoké učení technické v Brně). ISBN 80-214-0361-6.
- [18] LinuxCNC - Getting LinuxCNC [online]. [cit. 15. 03. 2018]. Dostupné z: http://linuxcnc.org/docs/2.7/html/getting-started/getting-linuxcnc.html#_installing_on_ubuntu_precise
- [19] User's Manual For M542 V2. Version 1.0, 2008 [online]. [cit. 21. 03. 2018]. Dostupné z: <http://www.cncshop.cz/PDF/M542.pdf>
- [20] LinuxCNC – Stepper Driver Timing [online]. [cit. 2. 04. 2018]. Dostupné z: http://wiki.linuxcnc.org/cgi-bin/wiki.pl?Stepper_Drive_Timing
- [21] HY-JK02-M 5-axis interface board manual [online]. [cit. 13. 04. 2018]. Dostupné z: <http://www.mantech.co.za/datasheets/products/1036147-R0%5E1.pdf>
- [22] LinuxCNC – Axis GUI [online]. [cit. 23. 04. 2018]. Dostupné z: <http://linuxcnc.org/docs/html/gui/axis.html>
- [23] Autodesk - CAM Reference [online]. [cit. 30. 04. 2018]. Dostupné z: <http://help.autodesk.com/view/fusion360/ENU/?guid=GUID-876FEBF5-2B91-46C2-B9A5-23542418037B>

Seznam symbolů, veličin a zkratek

r	-	Poloměr [m]
v	-	Rychlost [m/s]
a	-	Zrychlení [m/s ²]
m	-	Hmotnost [kg]
g	-	Tíhové zrychlení [m/s ²]
s	-	Stoupání závitu [m]
φ	-	Úhel natočení [rad]
ω	-	Úhlová rychlost [rad/s]
ε	-	Úhlové zrychlení [rad/s ²]
M	-	Moment síly [N · m]
F	-	Síla [N]
J	-	Moment setrvačnosti [kg · m ²]
FEKT	-	Fakulta elektrotechniky a komunikačních technologií
VUT	-	Vysoké učení technické v Brně
KM	-	Krokový motor
HW	-	Hardware
SW	-	Software

Seznam příloh

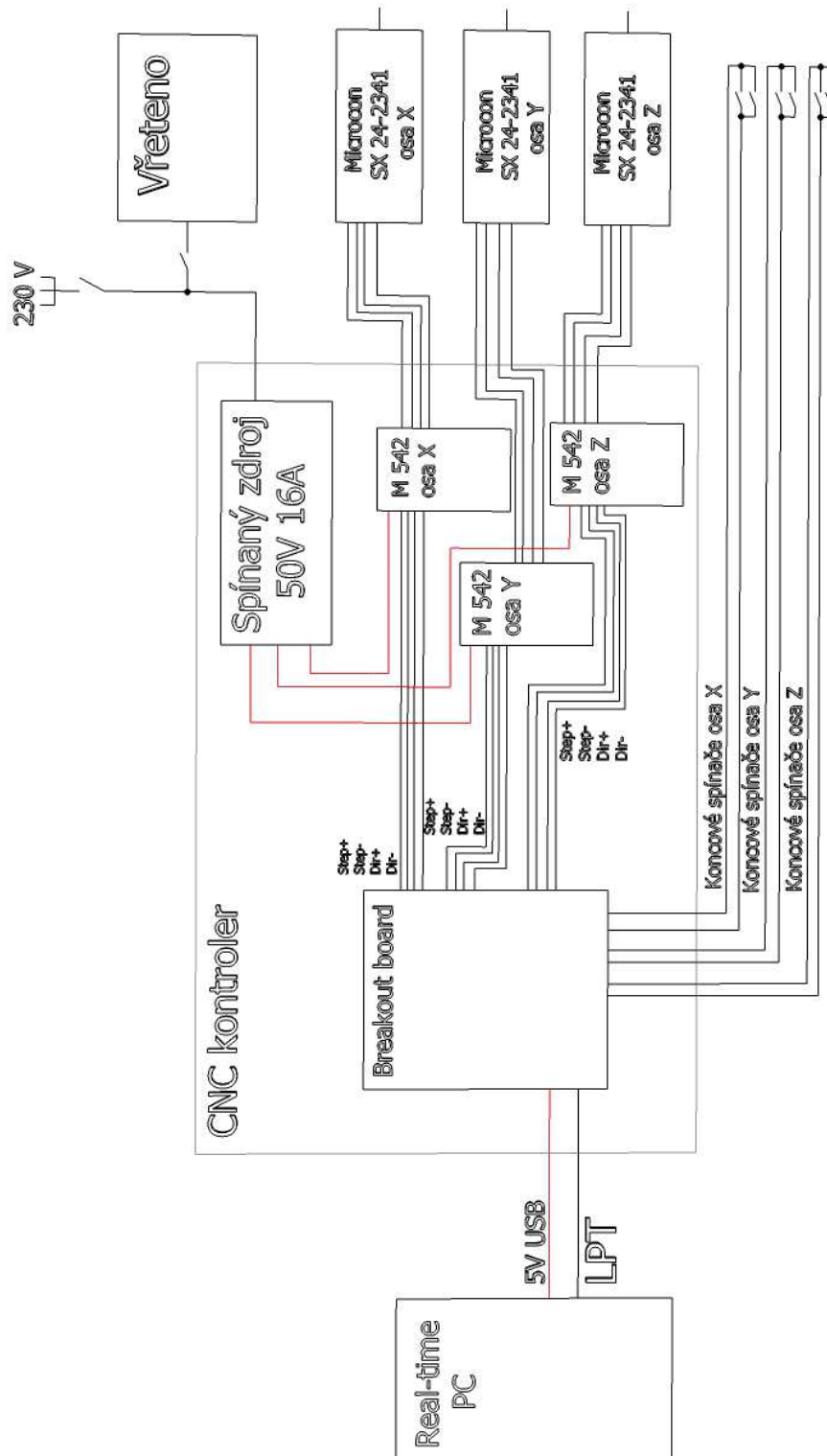
Příloha A: Schéma zapojení řídicí elektroniky.....	66
Příloha B: Aktuální podoba CNC frézky.....	67
Příloha C: CNC kontroler.....	68
Příloha D: CNC frézka – ovládací „setup“.....	68
Příloha E: Dokončené frézování – příchytné body.....	69
Příloha F: Hotový výrobek – dřevěný hlavolam.....	69

Seznam příloh na CD

- Elektronická verze bakalářské práce v PDF
- Kompletní fotodokumentace stroje, zapojení a prvních výrobků
- Videozáznam z obrábění
- Inicializační a konfigurační soubory aktuálního nastavení LinuxCNC
- Soubor s upraveným G-kódem pro výrobu dřevěného hlavolamu

Příloha A

Schéma zapojení řídicí elektroniky



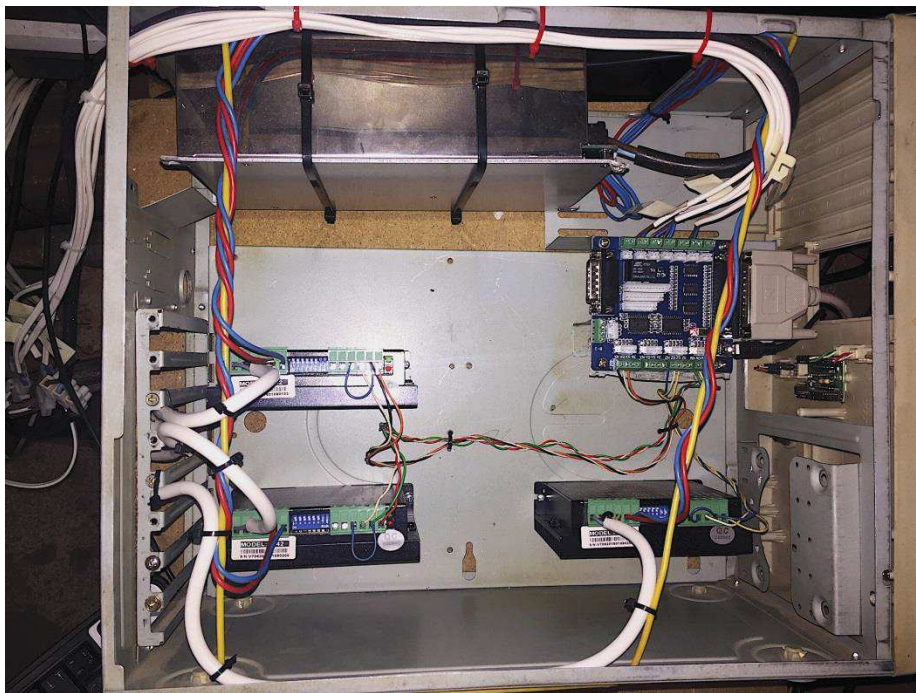
Příloha B

Aktuální podoba CNC frézky



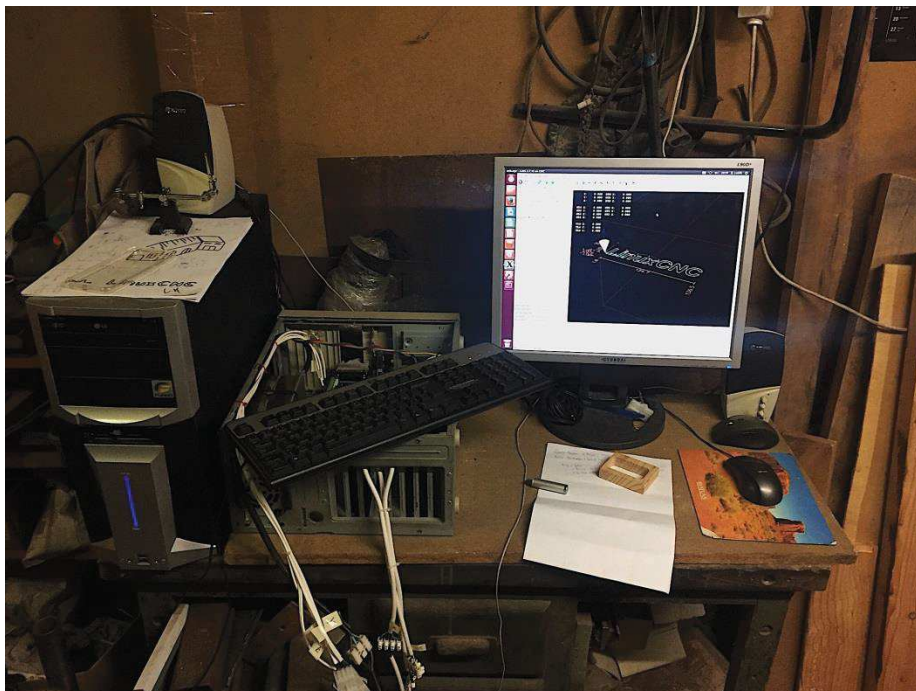
Příloha C

CNC kontroler



Příloha D

CNC frézka – ovládací „setup“



Příloha E

Dokončené frézování – příchytné body



Příloha F

Hotový výrobek – dřevěný hlavolam

