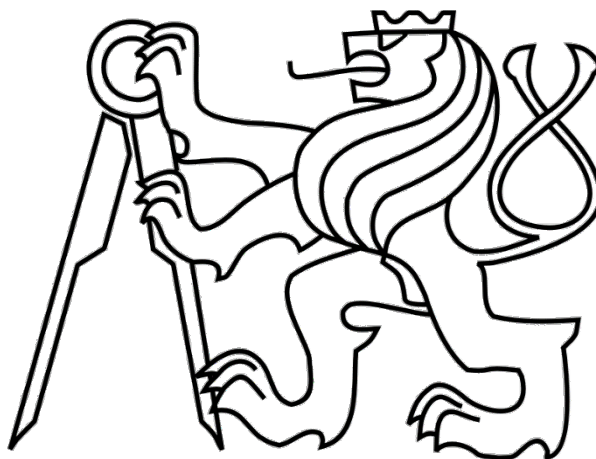


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

Ústav výrobních strojů a zařízení



Diplomová práce

Retrofit řídicího systému 3D tiskárny

2021

Bc. Jan Krejčí



ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Krejčí** Jméno: **Jan** Osobní číslo: **408795**
Fakulta/ústav: **Fakulta strojní**
Zadávající katedra/ústav: **Ústav výrobních strojů a zařízení**
Studijní program: **Průmysl 4.0**
Studijní obor: **bez oboru**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Retrofit řídicího systému 3D tiskárny

Název diplomové práce anglicky:

Retrofit of the 3D printer control system

Pokyny pro vypracování:

Popis tématu: Omezená životnost elektrotechnických komponent je příčinou ukončení provozu 3D tiskárny Stratasys s technologií tisku FDM. Mechanická konstrukce je v dobrém technickém stavu a svým nadčasovým provedením odpovídá konstrukci dnešních strojů. Za předpokladu instalace nového řídicího hardwaru a softwaru je možné zařízení uvést do funkčního stavu. Cílem práce je představit pohled možných systémů řešení elektronického hardwarového vybavení včetně definice ovládání a realizace řízení na stroji prostřednictvím poskytnutých komponent Beckhoff. ; Osnova práce: 1. Připravte rešerši parametrů a systémů určených pro řízení 3D tiskových strojů s technologií FDM., 2. Navrhněte koncept pro řízení 3D tiskového stroje Stratasys prostřednictvím elektronického hardwaru Beckhoff., 3. Realizujte softwarové oživení stroje s odpovídající vizualizací a ovládáním na řídicím panelu. ; Rozsah grafické části: Graficky bude na odpovídajícím formátu zpracováno blokové schéma realizovaného softwarového řešení. ; Rozsah textové části: 60-80 stran;

Seznam doporučené literatury:

1. SCHMID, Dietmar, et al. Řízení a regulace pro strojírenství a mechaniku. Praha : EUROPA-Sobotálesv cz, 2005. 420 s. ISBN 80-86706-10-9.
2. SOUČEK, Pavel. Pohony výrobních zařízení: Servomechanismy. Praha 1: ČVUT, 1997.
3. MAREK, Jiří. Konstrukce CNC obráběcích strojů. 2. vyd. Praha 10: MM publishing, 2010. ISBN 978-80-254-7980-3.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Jan Koubek, Ph.D., ústav výrobních strojů a zařízení FS

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

Ing. Štěpán Chládek, Ph.D., ústav výrobních strojů a zařízení FS

Datum zadání diplomové práce: **30.04.2021**

Termín odevzdání diplomové práce: **25.07.2021**

Platnost zadání diplomové práce: **30.09.2021**

Ing. Jan Koubek, Ph.D.
podpis vedoucí(ho) práce

Ing. Matěj Sulitka, Ph.D.
podpis vedoucí(ho) ústavu/katedry

prof. Ing. Michael Valášek, DrSc.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

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

30.9.2021

Datum převzetí zadání

62

Podpis studenta

Prohlášení

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a že jsem uvedl v příloženém seznamu veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací, vydaným ČVUT v Praze 1. 7. 2009.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu § 60 Zákona č.121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze dne 25. 7. 2021

podpis_____

Poděkování

Velice děkuji Ing. Janu Koubkovi, Ph.D. a Ing. Štěpánu Chládkovi, Ph.D. za odborné vedení, technické konzultace a poskytnutí hardwaru potřebného pro tvorbu této diplomové práce.

Anotace

Autor:	Bc. Jan Krejčí
Název diplomové práce:	Retrofit řídicího systému 3D tiskárny
Rozsah práce:	100 stran
Školní rok vyhotovení:	2021
Škola:	ČVUT – Fakulta Strojní
Ústav:	U12108.2 – Ústav výrobních strojů a zařízení
Vedoucí diplomové práce:	Ing. Jan Koubek, Ph.D.
Konzultant:	Ing. Štěpán Chládek, Ph.D.
Zadavatel:	ČVUT – FS
Využití:	Tvorba softwaru pro ovládání 3D tiskárny na základě NC programu.
Klíčová slova:	3D tisk, krokové motory, Stratasys, Beckhoff, TwinCat
Anotace:	Diplomová práce se zabývá konceptem řízení 3D tiskáren. U 3D tiskárny Stratasys Dimension Elite/768 má dojít k retrofitu původního řídicího systému. Původní řešení 3D tiskárny nahradí řídicí systém Beckhoff, který obslouží jednotlivé funkce 3D tiskárny v závislosti na použitých ovládacích modulech průmyslového počítače Beckhoff. Tvorba software potřebného pro ovládání krokových motorů a načtení NC programu je vytvořena v programu TwinCat XAE Shell. Výsledkem je tiskárna, která má funkční všechny tři pohybové osy a dokáže vykonávat interpolovaný pohyb na základě NC programu. Pro ovládání slouží HMI ze kterého lze ovládat osy, najíždět do referenčních poloh, a načítat NC program.

Annotation

Author:	Bc. Jan Krejčí
Title of a master's thesis:	Retrofit of the 3D printer control system
Extent:	100 pages
Academic year:	2021
University:	CTU in Prague, Faculty of Mechanical Engineering
Department:	U12108.2 – Department of Production Machines and Equipment
Supervisor:	Ing. Jan Koubek, Ph.D.
Consultant:	Ing. Štěpán Chládek, Ph.D.
Submitter of the theme:	CTU, Faculty of Mechanical Engineering
Application:	Programing software for control of 3D printed on the basis of NC program.
Key words:	3D print, stepper motors, Stratasys, Beckhoff, TwinCat
Annotation:	<p>The diploma thesis deals with the concept of control of a 3D printer Stratasys. The original 3D printer solution will replace the Beckhoff control system, which handles the individual functions of the 3D printer depending on the control modules used in the Beckhoff industrial computer. The software required to control the stepper motors and load the NC program is created in the TwinCat XAE Shell program. The result is a printer that has all three motion axes functional and can perform interpolated motion based on an NC program. The HMI is used for control, from which the axes can be controlled, approached to reference positions, and the NC program loaded.</p>

Obsah

1. Úvod a cíle práce	11
2. Řídící systémy	12
2.1. PLC	13
2.2. Mikrokontrolér	14
2.3. Řídící systémy pro NC úlohy.....	16
3. Aditivní technologie.....	18
3.1. 3D Tisk - FDM.....	18
3.2. SLS	23
3.3. SLA	24
3.4. LOM	25
4. Konstrukční uspořádání pohybových os	26
4.1. Kartézské uspořádání	26
4.2. Delta.....	27
4.3. SCARA.....	28
4.4. H-BOT	28
5. Pohony.....	29
5.1. Krokové motory.....	30
Krokový motor s permanentním magnetem	31
Krokový motor s variabilní reluktancí - VR	32
Hybridní krkový motor	33
Unipolární a bipolární krokové motory.....	35
5.2. Interpolovaný pohyb více pohonů (1).....	36
6. Retrofit 3D tiskárny Stratasys Dimension Elite/768	40
6.1. Řízení tiskárny	41
Dual-port RAM	41
Ovládání pohybových os X, Y, Z.....	41
Ovládání motoru posunu filamentu	42
Regulace teploty tiskové komory	42
Regulace teploty trysky	43
Rozvaděčová skříň.....	43



6.2. Řídící systém Beckhoff	45
TwinCat.....	46
IEC 61131-3.....	46
6.3. Propojení tiskárny Stratasys a řídicího systému Bekchoff	48
Koncové spínače	49
7. Tvorba programu.....	50
System	51
Motion	51
PLC	52
I/O	53
7.1. Popis funkce programu	54
7.2. Řízení os 3D tiskáren.....	56
Řízení krokových motorů	57
Mikrokrokování.....	57
Nastavení polohování	60
7.3. NC program.....	65
7.4. Popis struktury programu:	67
7.5. Uživatelské rozhraní.....	69
7.6. Potřebné úkony a hardware pro navazující zprovoznění	71
8. Závěr.....	73
9. Seznam obrázků	74
10. Seznam použité literatury	76
11. Přílohy	79

Seznam použitých veličin, jednotek

m	Počet vinutí motoru	[-]
N_r	Počet rotorových zubů	[-]
n	Otáčky motoru	[s ⁻¹]
v	Rychlost posuvu	[mm/s]
φ	Velikost jednoho kroku	[°]
$x_{\check{z}}$	Žádaná poloha v ose x	[mm]
$y_{\check{z}}$	Žádaná poloha v ose y	[mm]
t	Čas	[s]
v_0	Rychlost	[mm/s]
K_{vx}	Rychlostní zesílení v ose X	[-]
K_{vy}	Rychlostní zesílení v ose Y	[-]
ε	Polohová odchylka	[mm]
R_0	Požadovaný poloměr	[mm]
ΔR	Polohová odchylka	[mm]
x_{skut}	Skutečná poloha v ose x	[mm]
y_{skut}	Skutečná poloha v ose y	[mm]
\ddot{y}	Zrychlení v ose y	[mm/s ²]
\ddot{x}	Zrychlení v ose x	[mm/s ²]
V_{AB}	Rychlost AB	[mm/s]

Seznam zkratk

CPU	Centrapl processing unit
3D	Ttree dimensional
ČVUT	České vysoké učení technické
PLC	Programmable logic controler
CNC	Computer numerical control
NC	Numerical control
FDM	Fused Deposition Modeling
PID	Proportional-Integral-Derivate
SLS	Selective Laser Sintering
SLA	Stereolitography
LOM	Laminated Object Manufacturing
SCARA	Selective Compliance Assembly Robot Arm

1. Úvod a cíle práce

3D tiskárny dnes nachází uplatnění téměř v každém odvětví průmyslu. Může jít o výrobu prototypových částí, či produkci tvarových součástek, které je obtížné vyrobít konvenčními metodami obrábění. Funkční výrobní zařízení je tak velmi užité pro průmyslové provozy, vývojové laboratoře či technické vzdělávací instituty. V inventáři Ústavu výrobních strojů a zařízení Fakulty strojní ČVUT v Praze je FDM 3D tiskárna Stratasys Dimension Elite/768. Mechanické provedení 3D tiskárny je velmi kvalitně provedeno, což zaručuje mnoho let spolehlivého fungování. Originální řídicí systém nicméně tak dlouhou životnost nemá a jeho oprava či výměna servisní službou společnosti Stratasys je velmi nákladná. Nabízí se tak možnost použití soudobého řídicího systému, který dokáže obsloužit všechny funkce, které jsou pro fungování 3D tiskárny podstatné.

Rešerše této práce se v počátku zabývá samotnými řídicími systémy pro výrobní stroje, dále obsahuje stručný popis nejpoužívanějších technologií aditivní výroby, konkrétně FDM, SLA, SLS, LOM. Primárně nás zajímá technologie FDM, pro níž existuje na trhu více variant konstrukčního uspořádání pohybových os, což má vazbu na způsob a obtížnost řízení pohonů pro dosažení požadované pozice tiskové hlavy.

Následující kapitola se zabývá krokovými motory a aktuálními možnostmi řídicích systémů pro jejich ovládání. Poslední kapitolou teoretické části diplomové práce je popis FDM 3D tiskárny Stratasys, které se retrofit řídicího systému týká. Pro cíle této práce jsou podstatné krokové motory pro řízení pohybových os a referenční snímače koncových poloh.

Cílem praktické části je zprovoznění interpolovaného pohybu tří pohybových os 3D tiskárny Stratasys, realizovaného pomocí řídicího systému Beckhoff. Je třeba vytvořit program v prostředí TwinCat v jazyce IEC-61311, jehož pomocí lze řídit osy manuálně a každou zvlášť. Dále je možné použít automatický režim, který umožňuje (1) najetí os do referenčních poloh a (2) vykonávání interpolovaného pohybu na základě NC programu v G-kódu. Pro komfortní ovládání je třeba vytvoření uživatelského grafického rozhraní, umožňující aktivaci jednotlivých programovaných funkcí. Kromě vytvoření řídicího programu je nutné nastavit parametry pro řízení krokových motorů pohybových os 3D tiskárny.

2. Řídicí systémy

Pro obsluhu funkcí ať už 3D tiskárny, nebo jiného výrobního zařízení, je třeba použít řídicí systém. Dříve se mnohé procesy řídily na základě logických funkcí, které měly svou fyzickou reprezentaci v elektrotechnické komponentě. Dnes má v sobě každý řídicí systém či počítač mikroprocesor – CPU, součástku postavenou na velmi malých tranzistorech. Vyrábí se mikroprocesory, kde je velikost jednoho tranzistoru pouhých 7 nm. Primární rozdíly jsou v architektuře procesoru, podpoře operačního softwaru, stupni integrace a počtu jader. Pro řízení průmyslových zařízení se používají programovatelné logické kontroléry – PLC. Trh s řídicími systémy pro výrobní stroje se dá rozdělit na 3 skupiny. Prvním případem je, že společnost, která výrobní stroj vyrábí si buď sama, nebo pomocí dalšího subjektu vytvoří řídicí jednotku na míru pro dané zařízení. Zde se předpokládá, že se daný stroj bude vyrábět sériově, aby se vrátily náklady na vývoj řídicího systému. Další možností je použít PLC řídicích systémů, které se specializují na výrobní stroje. Aktuálně nejpoužívanější jsou řídicí systém Sinumeric od společnosti Siemens a řídicí systém HAIDENHAIN od stejnojmenné společnosti. Tyto systémy se používají kvůli své ceně především na větších CNC výrobních strojích. Jsou to patrně ty nejspolehlivější a nejpropracovanější systémy, které lze pro řízení výrobních strojů použít.



Obrázek 1: Ovládací panel řídicího systému Sinumeric (5)

Další možností je použít modulární PLC systém, který umožňuje řízení pohonů, dokáže obsloužit všechny periferie daného výrobního stroje a zároveň má vývojové prostředí ve kterém lze požadované úkony naprogramovat, případně jsou součástí knihovny funkcí. Takové řešení nabízí například společnosti Festo, Mitsubishi či Beckhoff.

2.1. PLC

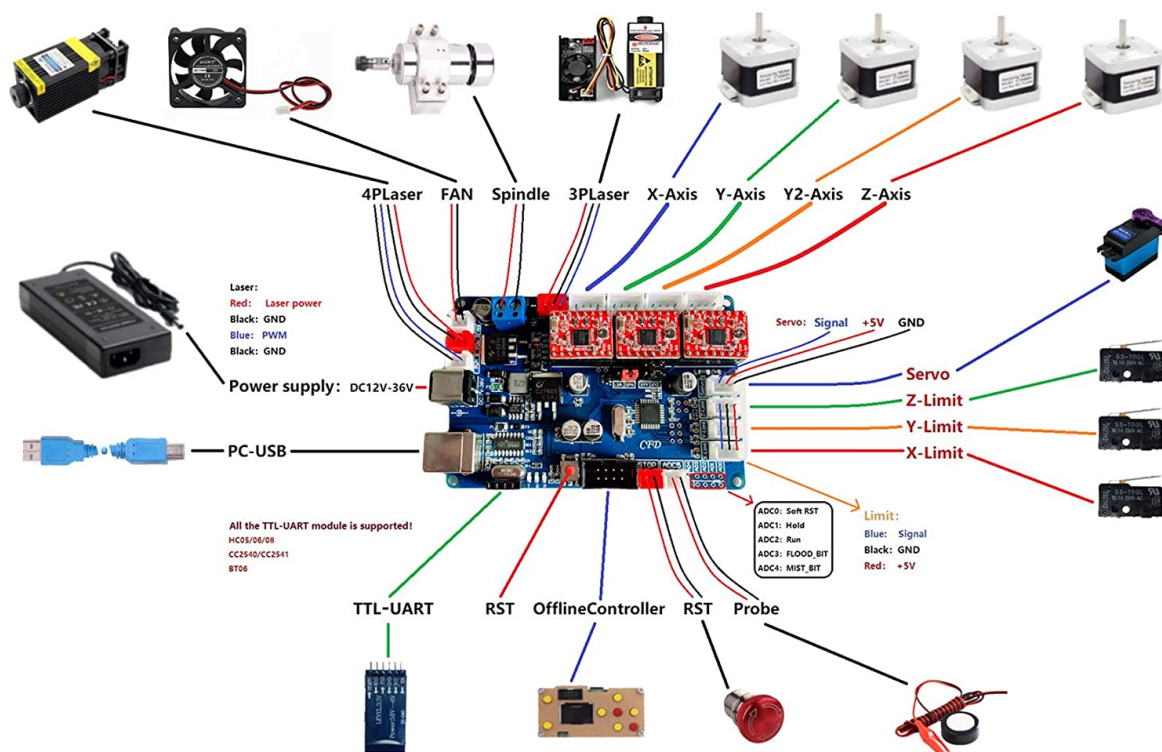
PLC neboli Programmable Logic Controller je kompaktní průmyslový počítač určený k řízení systémových procesů z jednoho místa. Historicky je nástupcem reléových zařízení, první PL bylo vyvinuto pro General Motors v roce 1968. PLC se mohou pohybovat od malých modulárních zařízení s desítkami vstupů a výstupů (I/O) v krytu integrovaném do procesoru, až po velká modulární zařízení s tisíci I/O, která jsou často připojena k síti s jinými PLC a SCADA systémy. PLC se skládají z procesoru, I/O modulů a napájecího zdroje. Procesor provádí předprogramované řídicí funkce na základě dat přijatých z připojených vstupních zařízení, kterými jsou snímač, spínač nebo teploměr. I/O moduly PLC jsou fyzicky připojeny a poskytují vstupní data do procesoru. Procesor na základě vstupních dat a programu komunikuje se výstupními zařízeními. I/O moduly jsou analogové či digitální a lze je kombinovat a přizpůsobit tak, aby vyhovovaly konkrétní aplikaci.

Jeli PLC v režimu RUN, tedy v provozu, vykonává cyklicky svůj běh s periodou v řádu milisekund. Nejprve je vždy přečten ze systémové paměti stav vstupů, následně jsou hodnoty zpracovány programem a výstupní proměnné uloženy do výstupních registrů systémové paměti. Doba jednoho pracovního cyklu je tím větší, čím je program obsáhlejší a mikroprocesor pomalejší.

PLC jsou speciálně navrženy a důkladně testovány, aby bylo zajištěno, že vydrží provoz v průmyslovém prostředí, kde mohou být vystaveny nárazům a vibracím, hluku, korozivním materiálům a extrémním výkyvům teploty. Podobně jsou PLC konstruovány pro dlouhou životnost a mnoho výrobců provozuje stejné PLC ve svých zařízeních po více než deset let. To, že jsou PLC úspěšné napříč celým průmyslem je také dáno tím, že jsou spolehlivé, flexibilní, přizpůsobitelné a programovatelné. Co se týče programování, v minulosti neexistovaly žádné standardizované koncepce a metody programování PLC,

což přinášelo problémy jako špatná softwarová architektura, omezená kontrola pro provádění programu, žádná společná syntax a sémantika, žádná přenositelnost softwaru. Řešením je programovací jazyk IEC 61131, který standardizovala v roce 1993 Mezinárodní elektrotechnická komise. (2,3)

Třetí možností, jak řídit výrobní stroj je použití vývojové desky s mikrokontrolerem, příklad se zobrazením možných periférií je na obrázku 2. Používá se především u low-endových zařízení s tlakem na nízkou pořizovací cenu, což se týká většiny levnějších 3D tiskáren na trhu. Takové řešení není příliš robustní a v průmyslu, kde vyžadujeme maximální spolehlivost se tak téměř nevyskytuje.



Obrázek 2: Řídící deska postavená na systému Arduino (6)

2.2. Mikrokontrolér

Mikrokontroler je kompaktní zařízení s integrovaným obvodem, které je obvykle zabudováno do jiného zařízení. Mikrokontrolery obsahují procesor (CPU), datovou paměť, programovou paměť, ovládání vstupu / výstupu (I/O) a podpůrné obvody. Data z různých

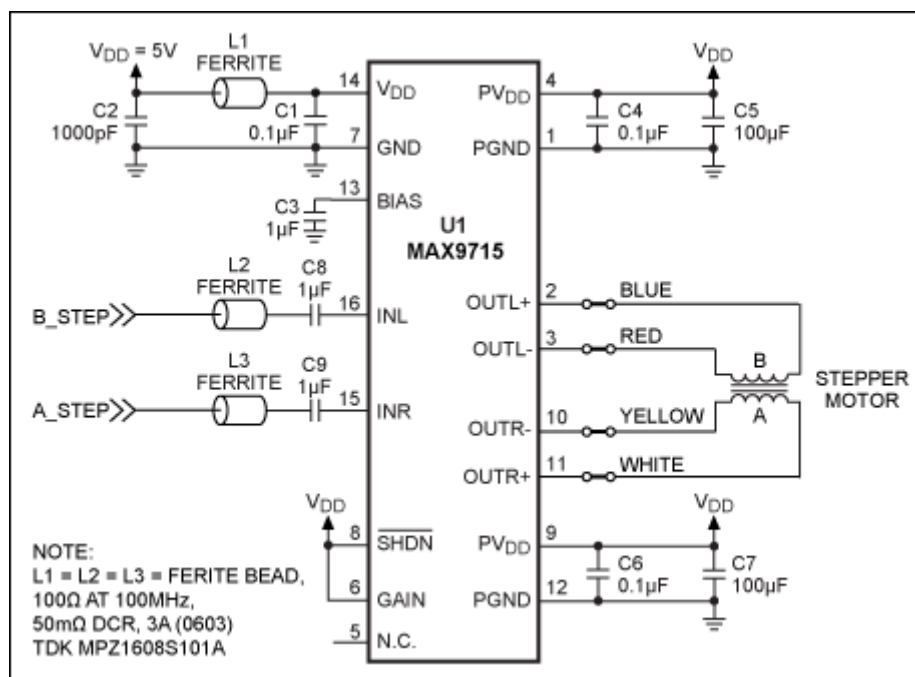
vstupů zařízení jsou přijímána do mikrokontroleru a dočasně ukládána do paměti. Procesor poté analyzuje data a určuje příslušnou odezvu na základě předem naprogramovaných pokynů uložených v programové paměti mikrokontroleru. Procesor poté použije I/O výstupy ke komunikaci odpovědi a provedení výstupní funkce.

Mikrokontrolery vynikají v aplikacích vyžadujících omezené výpočetní funkce. Mohou však provádět širokou škálu operací, včetně matematických výpočtů, zpracování logiky a dat, ovládání zařízení a zpracování sériové a bezdrátové komunikace. Mikrokontrolery jsou navíc obvykle kompaktní a velmi levné.

Sortiment mikrokontrolerů, označovaných také jako vývojové desky je obrovský. Zařízení jsou postavena především na systémech Audino či Rapsbery Pi. Vzhledem k tomu, že jsou tyto platformy open-source, jsou zdrojové kódy pro různé aplikace či dílčí úkoly volně k dispozici.

Na trhu jsou mikrokontrolery, které se zpravidla skládají ze dvou navzájem propojených desek. Na jedné je mikroprocesor a komponenty potřebné pro jeho fungování, na druhé desce je napájecí člen, tranzistory, drivery a další periférie. Mezi často používané vývojové desky patří Arduino Mega 2560 postavená na 8-bit mikroprocesoru ATMEGA 2560. 8-bitové procesory jsou nicméně kvůli nízkému výpočetnímu výkonu vytlačovány 16-bit a 32-bit mikroprocesory. Ty jsou ve velké míře založeny na CPU ARM Cortex-m3, který mimo větší paměti a nižší spotřebu elektrické energie nabízí i vyšší výpočetní výkon díky vyšší taktovací frekvenci. Kromě mikroprocesoru jsou pro správné fungování polohování 3D tiskárny důležité použité drivery. Ovladače krokových motorů mohou být součástí řídicí desky, což sebou nese tu nevýhodu, že nejdou nijak vyměnit či nahradit. V mnoha případech se však používají externí ovladače, které jsou v podobě malé desky připojené na pinové lišty řídicí desky. Základní částí driveru je integrovaný obvod, který na základě vstupních signálů generuje PWM signál, jehož pomocí jsou řízeny výkonové tranzistory. Schéma driveru je na obrázku 3.

Zásadní parametr driveru pro krokové motory je počet mikrokroků – na kolik dílčích kroků dokáže driver rozdělit základní krok, rozsah použitelného napájecího napětí, velikost trvalého a špičkového napětí.



Obrázek 3: Schéma driveru krokového motoru (7)

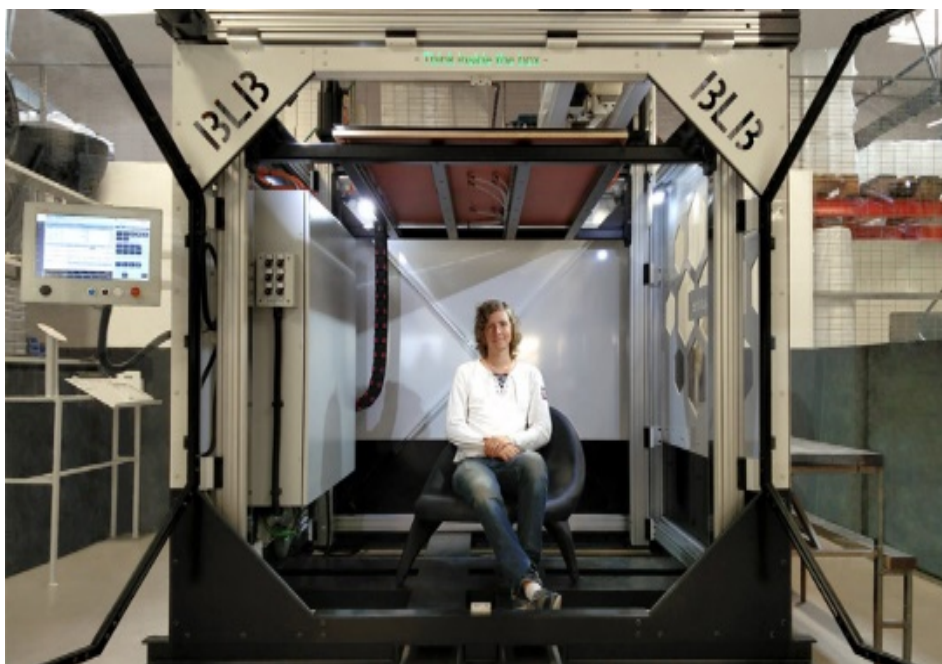
2.3. Řídicí systémy pro NC úlohy

Řízení pohybu tiskové hlavy je prováděno na základě NC programu v G-kódu. Součástí vhodného řídicího systému musí být interpret G-kódu, který zajistí správné přeložení G-kódu na příkazy pro drivery krokových motorů. Mezi PLC a mikrokontrolery je velký cenový rozdíl, ovšem pokud uvažujeme nasazení do průmyslové aplikace, je spolehlivost a robustnost mnohem důležitější než pořizovací náklady.

Výběr řídicího systému výrazně ovlivňuje druh pohonů, jenž chceme řídit. Pro střídavé servopohony jsou tu osvědčené systémy pro řízení CNC obráběcích strojů od společností jako jsou Siemens, Heidenhain či Fanuc. U 3D tisku jsou dynamické síly při výrobním procesu absolutně zanedbatelné v porovnání s obráběcími stroji. Řízení tak probíhá v otevřené regulační smyčce bez zpětné vazby v podobě enkodéru či lineárního odměřování.

Převážná většina 3D tiskových strojů má jako pohony krokové motory. Na trhu je široká škála samostatných driverů pro krokové motory, které jsou velmi užitečné, pokud chceme řídit rychlost, polohu i kroutící moment, což je vhodné pro jednoúčelové aplikace,

či PTP pohyb. Pro řízení pohonů 3D tiskárny je optimální řešení zařízení s implementovaným interpretem schopné řídit interpolovaný pohyb. Z hlediska 3D tiskáren na trhu výrazně převyšuje použití mikrokontrolerů nad PLC systémy, což je dáno předně cenou. Zastoupení velmi zkresluje rozmach 3D tiskáren pro hobby použití, u kterých přesnost výsledku není tak důležitá a neovlivní průmyslovou výrobu. Ty nejlevnější – Ender mají prodejní cenu od 5 000 Kč. Přelom do 3D tiskáren pro průmyslové použití začíná cca na 50 000 Kč. Cena plnohodnotného PLC s příslušnými kartami pro řízení motorů je 30 000 Kč a více. V případě 3D tiskárny za hypotetickou prodejní cenu 150 000 Kč, je to příliš velká část prodejní ceny, nicméně zajistí spolehlivý chod osvědčeného hardwaru. Na trhu jsou ovšem průmyslové 3D tiskárny, stavěné na zakázku, či 3D tiskárny s více tiskovými hlavami a velkými rozměry (Obrázek 8), u kterých se bez průmyslového PLC systému nelze obejít. Druhý případ pro použití PLC, tedy retrofit 3D tiskárny, se týká této práce. High-end zařízení, u kterého skončila životnost původního systému a má dojít k jeho nahrazení za ponechání stávající konstrukce a akčních komponentů.



Obrázek 4: The Box od společnosti BLB Industries (8)

3. Aditivní technologie

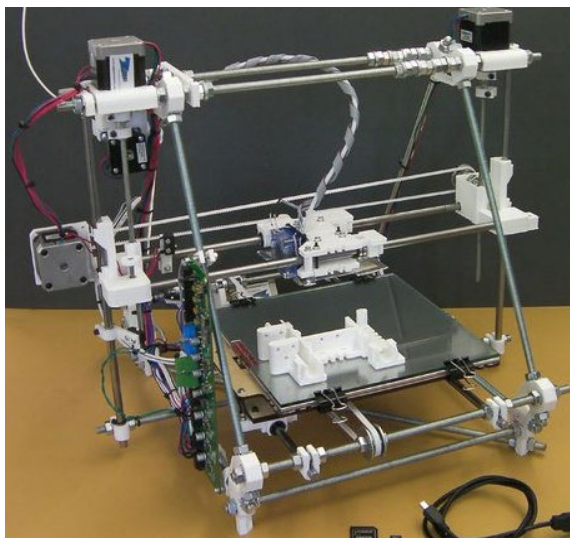
Na trhu je kladen důraz na stále rychlejší, levnější a kvalitnější výrobu. Zároveň se produkty vyrábí ve velkém množství variant a je třeba provádět rychlý vývoj u kterého jsou velmi užitečné aditivní technologie. Ty mají též za cíl zjednodušení výroby velice složitých výrobků. Je to technologie, při které se výrobek vytváří postupným vrstveným procesem.

Nejpoužívanějšími aditivními technologiemi jsou Fused Deposition Modeling, Selective Laser Sintering, Stereolitografie a Laminated Object Manufacturing. Všechny tyto technologie vyžadují řízený interpolovaný pohyb. Výběr řídicího systému pro technologii FDM, což je případ této práce je aplikovatelný i na ostatní popsané technologie.

3.1. 3D Tisk - FDM

3D tisk je aditivní technologie vytváření prostorových objektů pomocí 3D tiskáren. Princip spočívá ve vrstvení materiálu do požadovaného tvaru. Způsob, jakým dochází k nanášení jednotlivých vrstev se liší podle konkrétní technologie, která je závislá především na materiálu, ze kterého chceme výsledný produkt vytvořit.

Počátek 3D tisku datujeme do roku 1989, kdy S. Scott Crump s manželkou patentují po práci lepicí tavnou pistolí technologii Fused Deposition Modeling (FDM) a zakládají firmu Stratasys. FDM je dnes nejrozšířenější technologie 3D tisku s nejširším záběrem využití. Největší rozmach 3D tisku mimo průmyslové prostředí přinesl start projektu RepRap v roce 2005 (Obrázek 5). V loňském roce zasaženém pandemií bylo použito velké množství 3D tiskáren na části štítů pro pracovníky ve zdravotnictví, či sponky pro držení nasazeného respirátoru (obrázek 6).

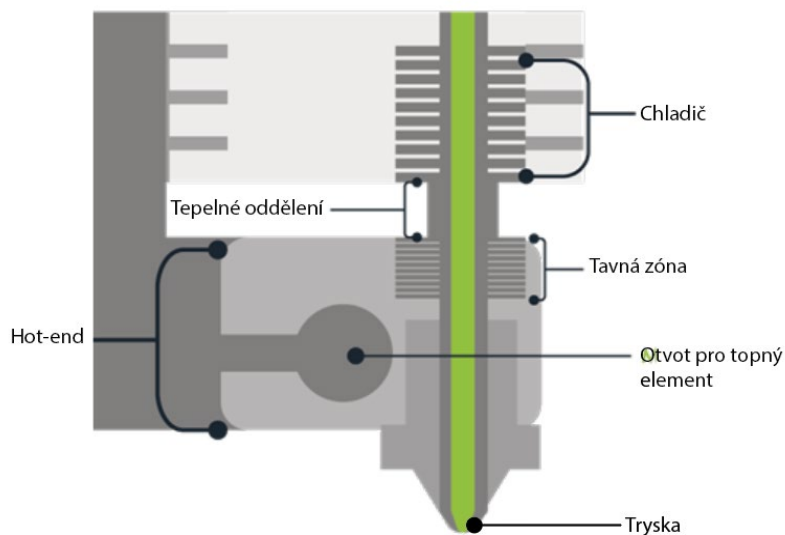


Obrázek 5: Tiskárna RepRap (9)



Obrázek 6: Štít pro pracovníky ve zdravotnictví (10)

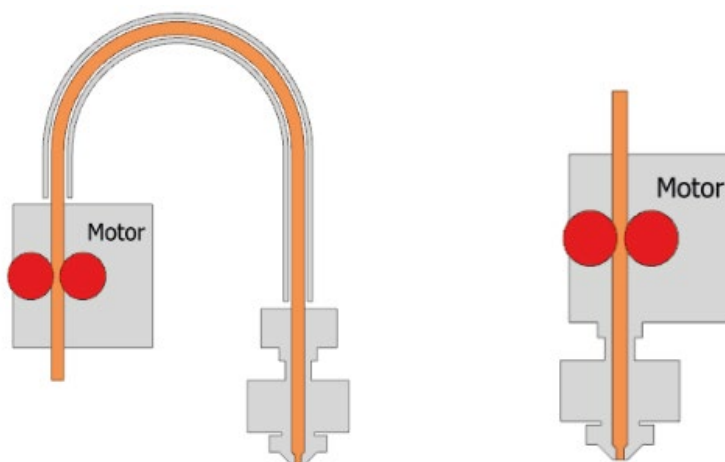
Samotná technologie FDM spočívá ve vrstvení nataveného polymeru. Základní částí FDM 3D tiskárny je extrudér – kovová část, ke které je připojen topný element a teplotní sonda. V extrudéru probíhá natavení polymerové struny – filamentu, a protlačení materiálu skrz trysku na určené místo v pracovním prostoru 3D tiskárny. Vnitřní průměr trysky extrudéru určuje, jak tenké stěny lze vytisknout a do určité míry i kvalitu tisku. Nákres extrudéru je na obrázku 7.



Obrázek 7: Extrudér FDM tiskárny (12)

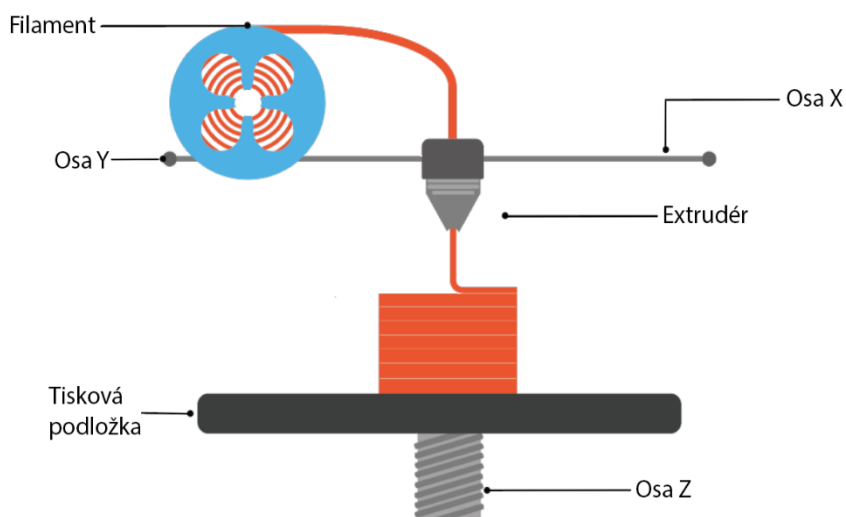
Extrudér se dále skládá z chladiče, který je od spodní části tzv. Hot-endu oddělen krčkem. Kdyby tomu tak nebylo, filament by se natavoval moc daleko od trysky, čímž následně dochází k ucpaní extrudéru.

Před extrudérem je umístěn mechanismus, který tlačí filament směrem k trysce. Posun je řešen jedním, či dvěma drážkovanými kolečky, která jsou umístěná proti sobě. Svírají filament, a díky přitlačné síle se do něj zatlačují a posouvají ho vpřed. Posuvný mechanismus je umístěn buď přímo nad extrudérem, nebo separátně a s extrudérem je propojen bowdenem ze silikonové trubičky s vnitřním průměrem o něco větším, než je vnější průměr filamentu, viz obr. 8.



Obrázek 8: Srovnání sestavy s bowdenem (vlevo) a bez bowdenu (13)

3D tiskárny se převážně konstruují jako stroje s kartézským souřadným systémem – tříosým, pravoúhlým (obrázek 6). Alternativní způsoby uspořádání pohybových os jsou rozebrány v kapitole 2.1.

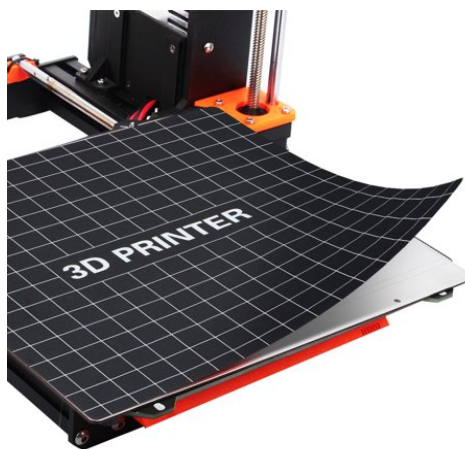


Obrázek 6: Schématický boční pohled na technologii FDM (14)

Natavený filament se nanáší na temperovanou tiskovou podložku. Povrch tiskové podložky může být vyroben z hliníku, nerezové oceli, skla, kaptonové folie, nebo polythermidu. Důležité je, aby byl materiál dobře tepelně vodivý a aby k němu natavený filament dobře přilnul. Zároveň nesmí být přilnutí příliš pevné. V takovém případě je velmi obtížné výtisk od podložky oddělit. V tomto ohledu mají výhodu podložky, které lze ohýbat, což je například pro sklo nerealizovatelné. V každém případě musí být tisková podložka rovná, aby byla zajištěna konstantní vzdálenost mezi tryskou a podložkou při tisku první vrstvy.

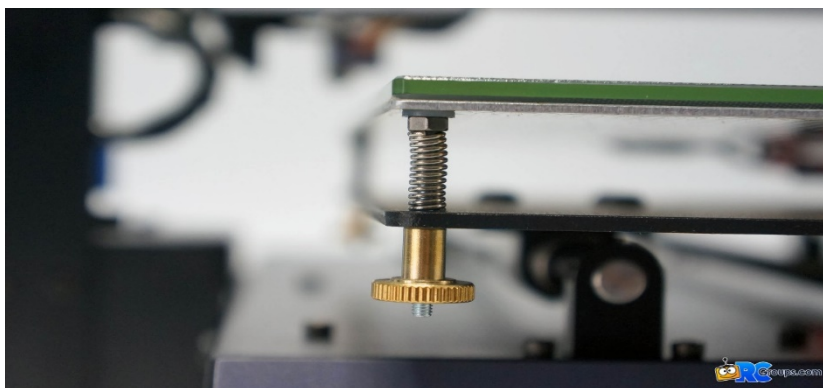
Tiskové podložky vyrobené z polythermidu mají dobrou smáčivost – tedy i přilnavost filamentu k povrchu podložky. Vrchní vrstva má drsnější povrch, čímž je přilnavost ještě lepší. Na tiskové podložky s velmi hladkým povrchem – sklo či ocel, je vhodné nanést lak, nebo lepidlo, které zajistí lepší přilnavost.

Pokud nemá tisková podložka, viz obrázek 7, vhodný povrch a tepelnou vodivost, může docházet k odlepování okrajů výtisku v průběhu výroby, nebo se v horším případě celý výtisk posune a je třeba začít znova. Podložka je umístěna na platformě s topnými tělesy, či topnou folií. Vhodné teploty pro většinu materiálů jsou 60° - 90°C . Na platformě je umístěn také teplotní senzor, který s PID regulátorem zajišťuje konstantní teplotu v průběhu celého tisku. Je důležité, aby měla podložka stejnou teplotu ve všech místech, tedy jak ve středu, tak na okrajích, čehož momentálně nejzajímavějším řešením je topná fólie rozdělená na více topných segmentů.



Obrázek 7: Tisková podložka (15)

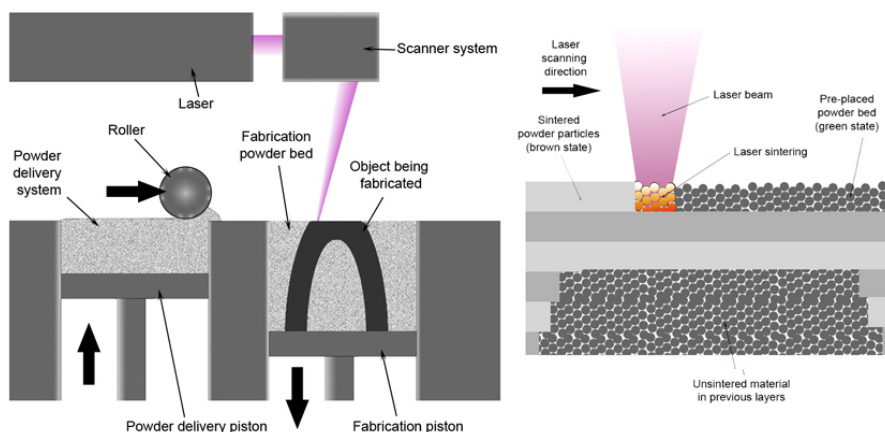
Aby byla zaručena konstantní vzdálenost mezi tryskou filamentu a tiskovou podložkou, je podložka umístěna na čtyřech kalibračních šroubech, kterými lze pozici a naklonění tiskové podložky doladit, viz obr. 8.



Obrázek 8: Kalibrační šroub (15)

3.2. SLS

Selective laser sintering, zkráceně SLS je technologie, kde se jako tiskové medium používá drcený prach a činitelem je výkonný laser. Používá se i termín sintrování. Technologie byla vyvinuta v osmdesátých letech. V pracovním prostoru je tisková podložka s osou Z. Na podložku je rolnou ze zásobní nádoby nanесena tenká vrstva drceného materiálu, může jím být kov, sklo, plast či keramika. Nad nádobou je umístěna skenovací hlava, která rozmítá v ose X a Y výkonný laserový paprsek. Účinkem paprsku je nataven drcený materiál a dojde tím k jeho spečení. Materiál, který zpevněn nebyl poslouží jako podpurný.



Obrázek 9: Nákres technologie SLS (16)

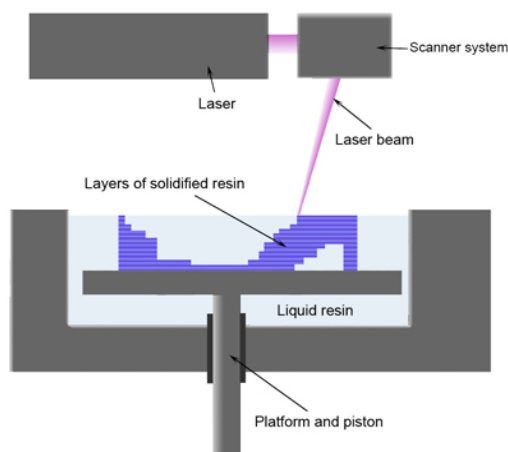
Tisková podložka se vznikajícím dílcem se následně posune o jednu vrstvu dolů a proces se opakuje. Nákres technologie SLS se nachází na obrázku č. 10 výše. Výrobek pomocí technologie SLS je pak znázorněn na obrázku následujícím.



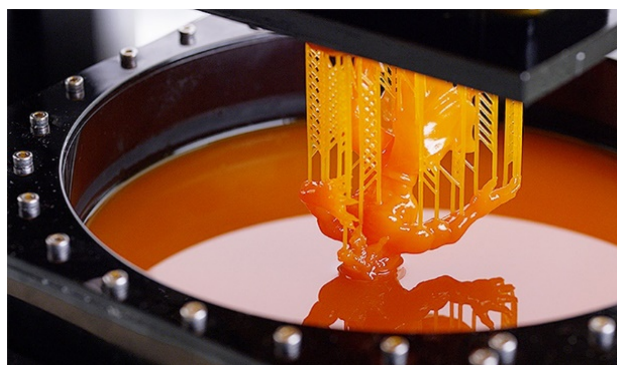
Obrázek 10: Dílec vyrobený technologií Selective Laser Sintering (19)

3.3. SLA

SLA je zkratka pro stereolitografii, což je technologie, která pro tvorbu výrobku využívá tekutého fotopolymeru a UV laseru. Stereolitografie byla patentována v roce 1986 společností 3D Systems. Fotopolymer je umístěn v nádobě, ve které je zároveň tisková podložka na ose Z. Nad nádobou je skenovací hlava s UV laserem, která rozmítá paprsek v ose X a Y. Po vytvrzení profilu dané vrstvy se tisková podložka posune o jednu vrstvu níže a probíhá vytvrzení další vrstvy. Tloušťka vrstvy se pohybuje od 0,05 mm do 0,15 mm. Pro vyrovnání rovinnosti je nutné před každou vrstvou přejet hladinou stěrkou. Technologií SLA lze oproti technologii FDM dosáhnout kvalitnějších výtisků (obrázek 11). Na druhou stranu není u SLA možná široká škála materiálů, které je možné použít. Pohyb podložky může být i obráceně, viz obrázek 11.



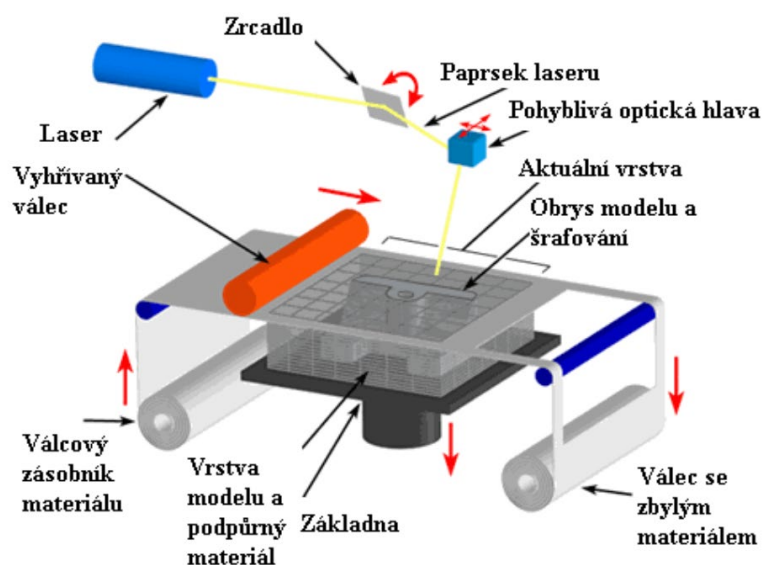
Obrázek 11: Nákres technologie SLA (16)



Obrázek 12: SLA výtisk (17)

3.4. LOM

Laminated Object Manufacturing, zkráceně LOM byla vyvinuta v roce 1988 společností Helysis. Dílec vzniká použitím speciálních plastových folií s tloušťkou 0,15 mm – 0,20 mm. Folie je navinuta na roli, ze které se odvíjí na pracovní plochu. Ta se posouvá v ose Z. Nad pracovní rovinou je laser, který do vrstvy vyřeže příslušný profil. Na vrstvu se nanáší lepidlo, pokud již není součástí folie. Podložka se posune, přisune se nový materiál a proces se opakuje. Přebytný materiál se rozřeže na čtvercové díly, aby šel výsledný produkt dobře vyjmout. Nevýhodou je, že přebytný materiál nelze vyjmout z dutin. Schématické zobrazení technologie s popisem je na obrázku 13.



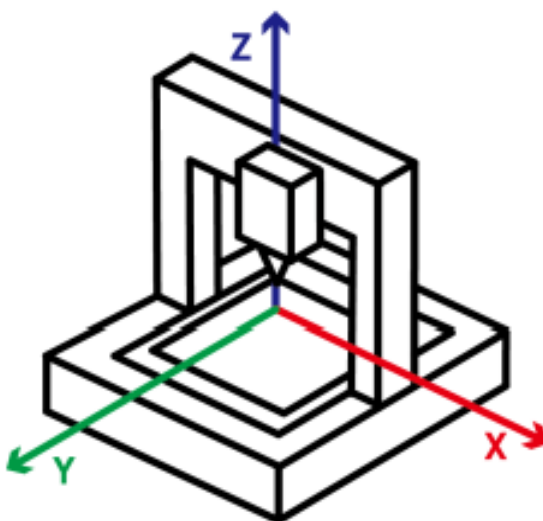
Obrázek 13: Technologie Laminated Object Manufacturing (LOM) (19)

4. Konstrukční uspořádání pohybových os

Primárně u FDM tiskáren je více konstrukčních možností, jak uspořádat prvky, díky kterým dostaneme trysku tiskárny na požadované souřadnice, což má vazbu na způsob řízení pohonů. Všechny níže uvedené konstrukční provedení pracují s 3-osým pravoúhlým souřadným systémem. Ne všechny ovšem dosahují požadované pozice posuvem ve směru souřadných os

4.1. Kartézské uspořádání

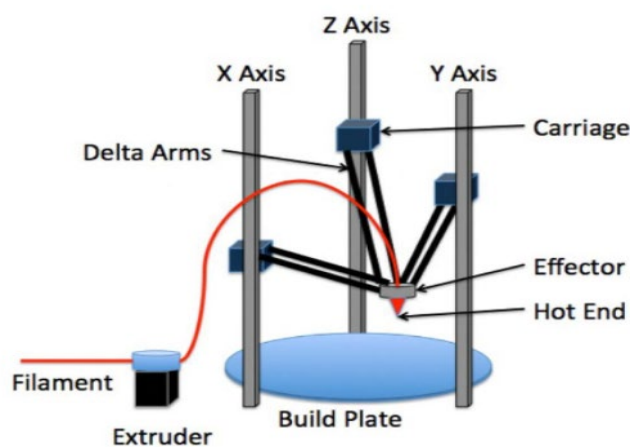
Kartézské konstrukční uspořádání (obrázek 14) je nejčastěji použité řešení. Tisková podložka se pohybuje ve směru osy Y, extrudéru náleží osa X a Z. Směr pohybových os souhlasí se směrem souřadného systému.



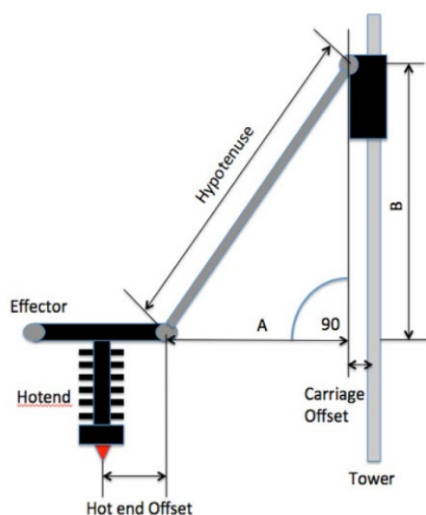
Obrázek 14: Kartézské konstrukční uspořádání os (26)

4.2. Delta

Delta tiskárny (obrázek 15 a 16) pracují se třemi nebo i více rameny připevněnými k vodorovným lineárním osám přes kulové čepy. Tisková hlava je spojena s koncem každého ramene. Pohyb pouzder na lineárních osách tak zajišťuje polohování tiskové hlavy v pracovním prostoru.



Obrázek 15: Schéma Delta tiskárny (27)



Obrázek 16: Geometrie ramene řešení Delta (27)

4.3. SCARA

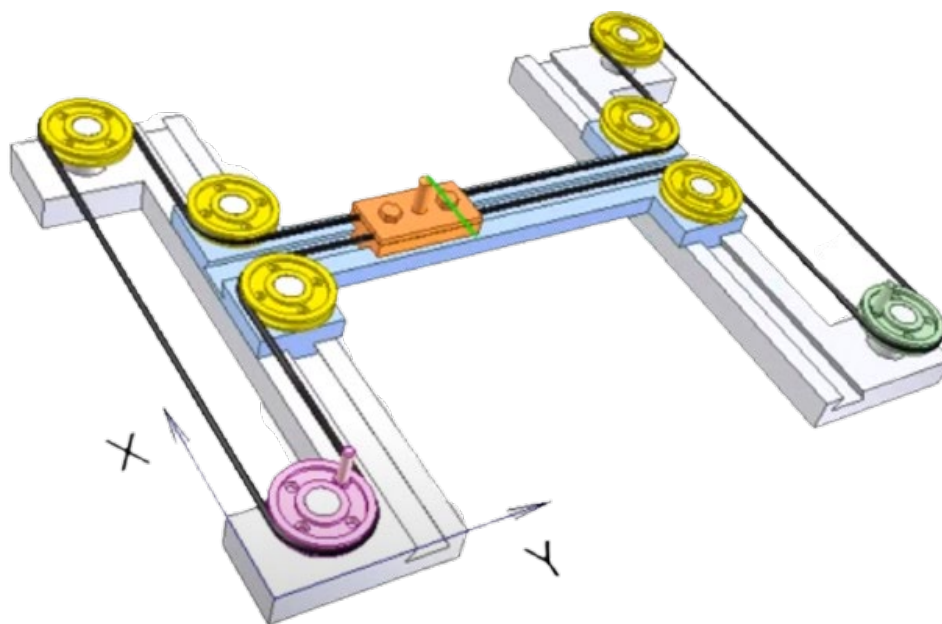
Zkratka SCARA znamená Selective Compliance Assembly Robot Arm. Jako přechodí konstrukční provedení využívá pravoúhlý kartézský souřadný systém, nicméně ovládané jsou v rovině XY pouze rotační vazby, jak je vidět na obrázku 17. Tisková hlava je umístěna na rameni, které má dvě části a dva rotační motory jako pohyb. Osa Z je obdobná jako u kartézské konstrukce.



Obrázek 17: Tiskárna s kinematikou SCARA (28)

4.4. H-BOT

Pohyb osy Z je zajištěn stejně jako u kartézské konstrukce. Pohyb v rovině XY je realizován pomocí jednoho řemenu, systému kladek a dvou hnaných kladek – na obrázku č. 18 je to zelené a růžové kolečko. Pokud se pohybuje pouze jedno z koleček, je dráha tiskové hlavy úhlopříčná vůči souřadnému systému roviny XY. Aby se pohybovala tisková hlava vodorovně s osou X nebo Y, je třeba součinnosti obou hnaných koleček.



Obrázek 18: H-bot – Kinematika os X a Y (29)

5. Pohony

Dle velikosti, pořizovací ceny a požadavků na přesnost a produktivitu stroje se na výrobních zařízeních používají různé typy pohonů. Na pomyslné špici jsou AC servopohony, které lze najít v každém CNC obráběcím stroji. Na výrobních strojích technologií aditivní výroby se v nemalé míře hlavně u menších strojů používají DC krokové motory. Jednak se díky mikrokrokování dobře řídí poloha a zároveň jsou krokové motory cenově efektivní variantou. Toto je případ i tiskárny Stratasys, které se tato práce týká.

5.1. Krokové motory

Krokový motor je točivý stroj, využívající elektromagnetické indukce a magnetismu, který se pohybuje nespojitě po jednotlivých úsecích – krocích, přičemž krok je mechanická odezva rotoru krokového motoru na jeden puls napájecího obvodu. Frekvence napájecích pulsů určuje rychlost otáčení motoru. Pohony s krokovými motory jsou vhodné jak pro přesné nastavování rychlosti, tak pro polohování. Mohou být použity jako polohová, případně rychlostní serva v otevřené zpětnovazební smyčce. Krokové motory existují již mnoho let. Jsou robustní, snadno ovladatelné a poskytují vysoký točivý moment. V mnoha aplikacích šetří zařízení pro počítání kroků drahé systémy zpětné vazby. I při stále častějším používání synchronních servomotorů krokové motory neztrácejí své místo na trhu. Považují se za vyspělou technologii a nadále se vyvíjejí s cílem snížit náklady a fyzickou velikost, zvýšit točivý moment a zlepšit spolehlivost. K mechanickému kontaktu, a tudíž otěru nedochází u krokových motorů jinde než v ložiskách. Vyznačují se proto velkou mechanickou odolností, dlouhou dobou života a provozem téměř bez údržby. Nevýhodou krokových motorů je tzv. ztráta kroků, která nastává při překročení mezního zatížení, a sklon k mechanickému zakmitávání, které může vést k nestabilitě při pohybu.

Základním parametrem krokového motoru je velikost kroku. Pro pootočení o jeden krok je třeba vybuzení jednoho ze statorových vinutí pro posun o jeden krok. Velikost jednoho kroku φ pro krokový motor s pasivním rotorem je dána rovnicí 1.

$$\varphi = \frac{360^\circ}{mN_r} \quad (1)$$

m – počet fází statoru

N_r – počet zubů rotoru

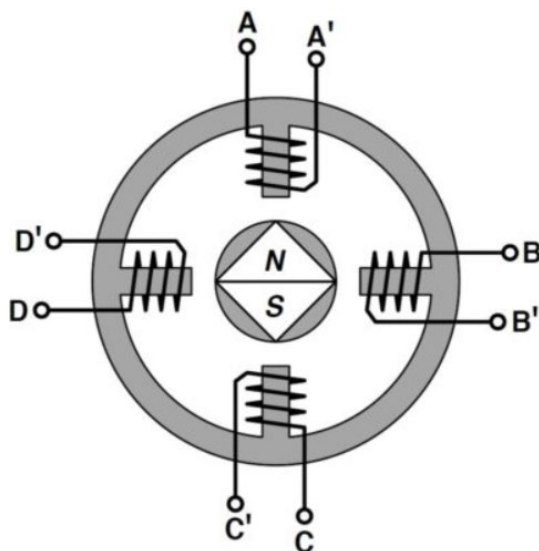
Krokové motory se vyrábějí jako jednofázové, nebo vícefázové. Polarita statorových pólů může být měněna dvěma způsoby. Je-li budící vinutí každého magnetu tvořeno jen jednou cívkou a přepólování je realizováno změnou směru proudu v cívce, mluvíme o bipolárním

provozu. Je-li každé vynutí tvořeno dvěma cívkami, mluvíme o unipolárním provozu. Každá cívka je zdrojem magnetického toku v jednom směru. Přepínáním cívek se mění polarita pólových párů statoru.

Krokové motory lze dělit dle více kritérií, nejdůležitějšími je konstrukční provedení, ve variantách variabilní reluktance, krokový motor s permanentním magnetem a hybridní krokový motor. Dalším důležitým rozdělením je způsob provedení vinutí, kde se krokové motory dělí na unipolární a bipolární. (30,31,32)

Krokový motor s permanentním magnetem

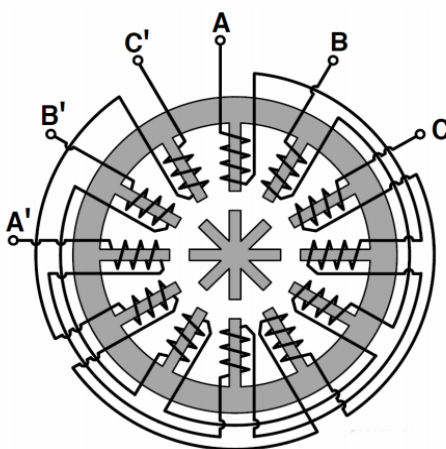
Na obvodu rotoru, vytvořeného z radiálně pólovaného permanentního magnetu, se u tohoto typu motoru střídají severní a jižní póly, jejichž počet je poloviční oproti počtu pólů statoru. Jednoduchá konstrukce těchto motorů (obrázek 20) a tím pádem nízké výrobní náklady je předurčují k použití ve spotřební a automobilové elektronice, kde není překážkou velikost jejich kroku ($7,5$ nebo 15°) a malý dosažitelný točivý moment.



Obrázek 20: Krokový motor s permanentním magnetem (30)

Krokový motor s variabilní reluktancí - VR

Nejstarší konstrukce krokových motorů, nazývaná též jako motor s pasivním rotorem, je dnes stále méně používaná. Stator i rotor jsou složeny z plechů s pólovými nástavci, na nichž jsou uložena vinutí jednotlivých fází a provedení je tří a vícefázové. Činnost je založena na změnách reluktance (poměr hnací síly a magnetického toku) magnetického obvodu stroje při otáčení rotoru. Dvojice cívek navinutých na protilehlých pólech je spojena a tvoří vždy jednu fázi krokového motoru.

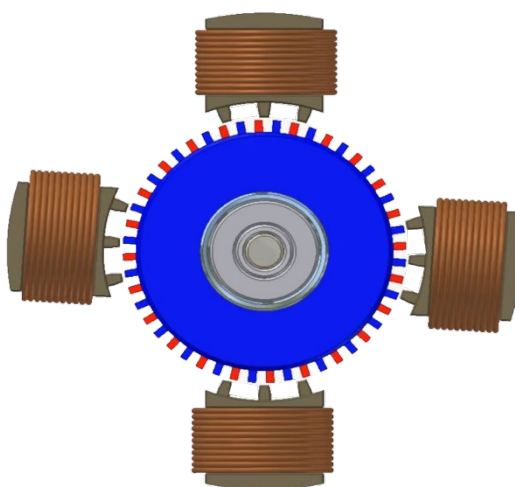


Obrázek 21: Variabilní reluktance zapojení (30)

Na obrázku č. 21 je zjednodušený řez magnetickým obvodem čtyřfázového krokového motoru s pasivním rotorem. Na statoru je 8 pólů, na kterých jsou navinuty cívky. Dvojice cívek navinutých na protilehlých pólech jsou spojeny a tvoří vždy jednu fázi krokového motoru. Tento krokový motor obsahuje celkem 4 fáze označené A, B, C a D. Rotor tvořený svazkem plechů na hřídeli má na svém povrchu 6 pólů a je bez vinutí. Fáze vinutí krokového motoru jsou připojeny na příslušné výstupy A, B, C, D ovladače.

Hybridní krkový motor

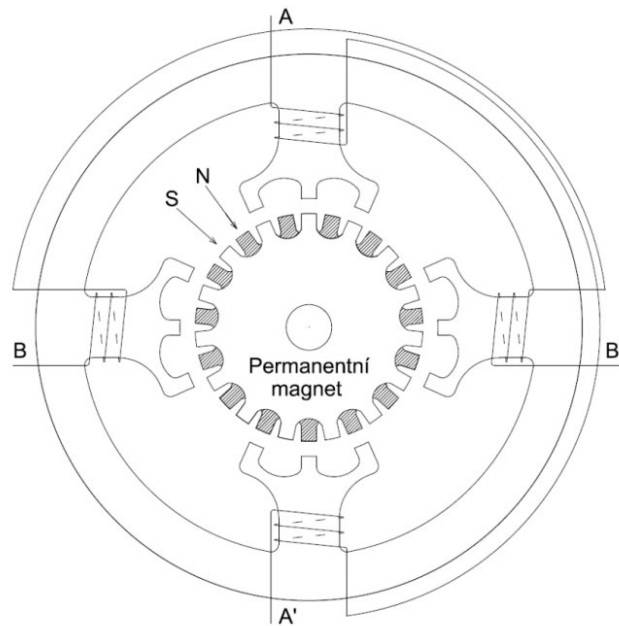
Základem hybridního krokového motoru je rotor tvořený z permanentního magnetu, na obou stranách je ozubení s čtvercovým profilem (Obrázek 22). Tyto ozubení jsou proti sobě pootočená o šířku jednoho zubu – jeden krok. Počet zubů udává počet kroků motoru. Motory této konstrukce vykazují nejlepší momentové i dynamické parametry a jsou dnes v průmyslových aplikacích používány téměř výhradně.



Obrázek 22: Rotor a stator hybridního krokového motoru (31)

Způsob, jakým dojde k otočení o jeden zub, je popsán pro dvoufázový krokový motor se čtyřmi pólovými nastavci, viz obrázek 22 a 23. Nejdříve je sepnuto vinutí fáze A, kde dojde k vzniku magnetického pole. Na pólu 1 bude jižní magnetická polarita přitahovat severní magnetický pól na rotoru. Pól 3 tím bude mít severní magnetickou polaritu a bude přitahovat jižní magnetický pól rotoru. Dalším krokem je sepnutí vinutí fáze B. Na pólu 2 je severní orientace a přitahuje jižní pól rotoru. Pól 4 pracuje v opačném magnetickém smyslu. Dojde tak k posunutí rotoru o čtvrt zubu doprava. Následuje krok 3, ve kterém je sepnuta fáze A v obrácené polaritě proti kroku 1. Dochází k přeorientování magnetických pólů, to znamená, že pól 1 zaujímá severní magnetickou polaritu a pól 3 jižní. Rotor se vychýlí tak, aby byla magnetické póly přitahovány a otočí se o další čtvrt otáčku. Čtvrtý krok nastává při sepnutí vinutí fáze B opačnou polaritou oproti kroku 2. Póly 2 a 4 mají opět

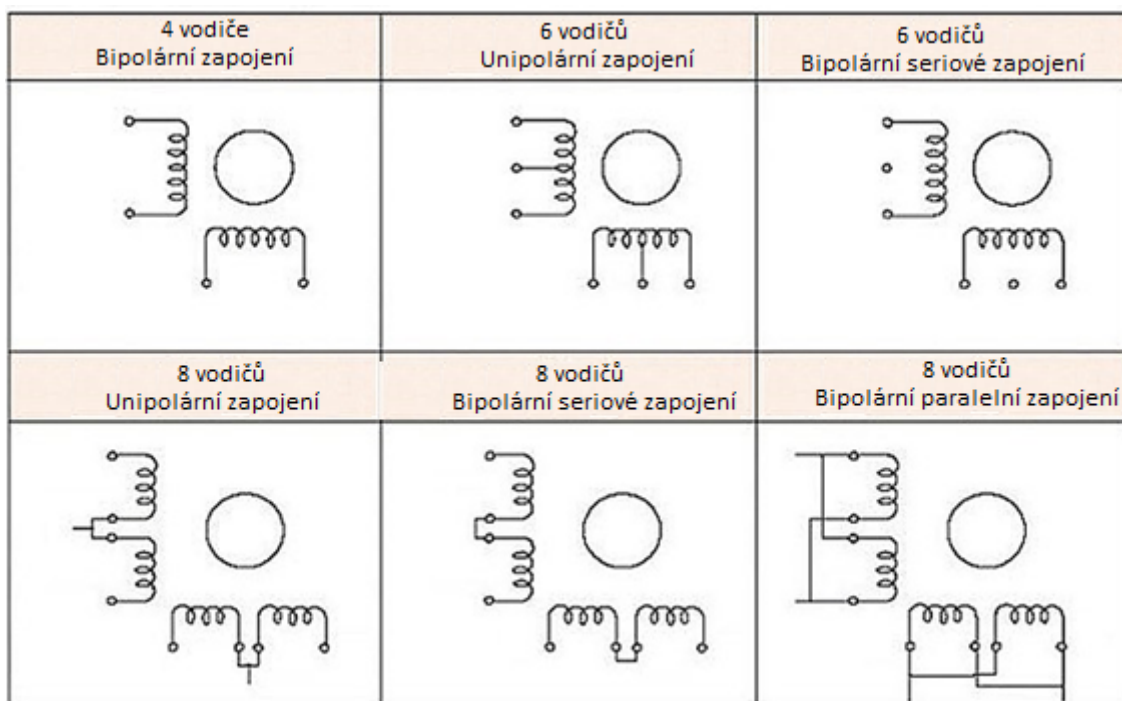
obráceně polarizované magnetické póly, které přitahují opačné magnetické póly rotoru.
 Následuje pootočení rotoru o čtvrtěční vzdálenost rotorových zubů.



Obrázek 23: Schéma hybridního krokového motoru (31)

Unipolární a bipolární krokové motory

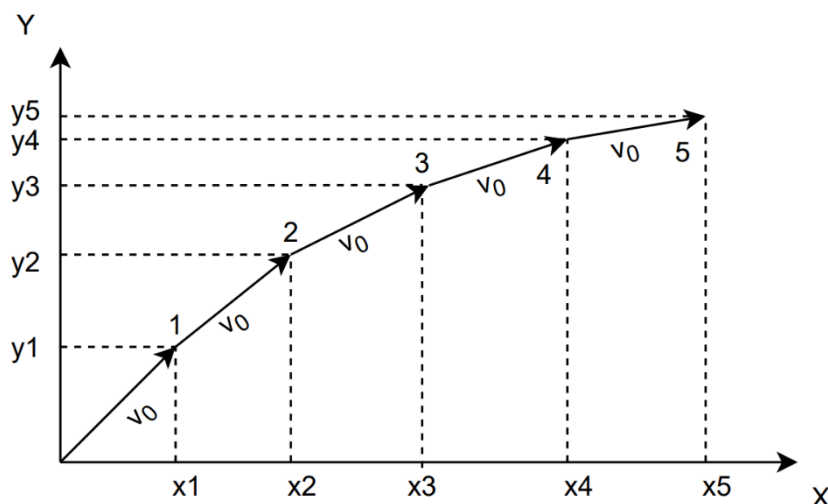
Krokové motory se vyrábějí jako jednofázové, nebo vícefázové. Polarita statorových pólů může být měněna dvěma způsoby. Je-li budící vinutí každého magnetu tvořeno jen jednou cívkou a přepólování je realizováno změnou směru proudu v cívce, mluvíme o bipolárním provozu. Je-li každé vynutí tvořeno dvěma cívkami, mluvíme o unipolárním provozu. Každá cívka je zdrojem magnetického toku v jednom směru. Přepínáním cívek se mění polarita pólových párů statoru. Porovnání zapojení cívek při unipolárním a bipolárním zapojení je na obrázku 24. (1,32)



Obrázek 20: Unipolární a bipolární zapojení (32)

5.2. Interpolovaný pohyb více pohonů

Pro správné polohování ve dvou a více osách, což vyžaduje součinnost více pohonů je třeba pohyb interpolovat. Požadovaná křivka je v případě lineární interpolace (obrázek 25) nahrazena přímkovými úseky a pohonům v obou osách x , y jsou zadávány rychlosti tak, aby byla dodržena výsledná posuvová rychlost v_0 (1).



Obrázek 21: Lineární interpolace v kartézském souřadném systému

Mezi body 1 a 2 bude vztah:

$$x_{\dot{z}} = x_1 + v_x t \quad (2)$$

$$v_x = v_0 \frac{x_2 - x_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \quad (3)$$

$$y_{\dot{z}} = y_1 + v_y t \quad (4)$$

$$v_y = v_0 \frac{y_2 - y_1}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}} \quad (5)$$

Rovnice žádané trajektorie pak bude mít následující tvar:

$$\varphi y_{\dot{z}} - y_1 = \frac{v_y}{v_x} (x_{\dot{z}} - x_1) \quad (6)$$

Pohony osy x y zaujmou při zahrnutí rychlostní konstanty skutečné souřadnice dle následujících rovnic.

$$x = x_z - \frac{v_x}{K_{vx}} \quad (7)$$

$$y = y_z - \frac{v_y}{K_{vy}} \quad (8)$$

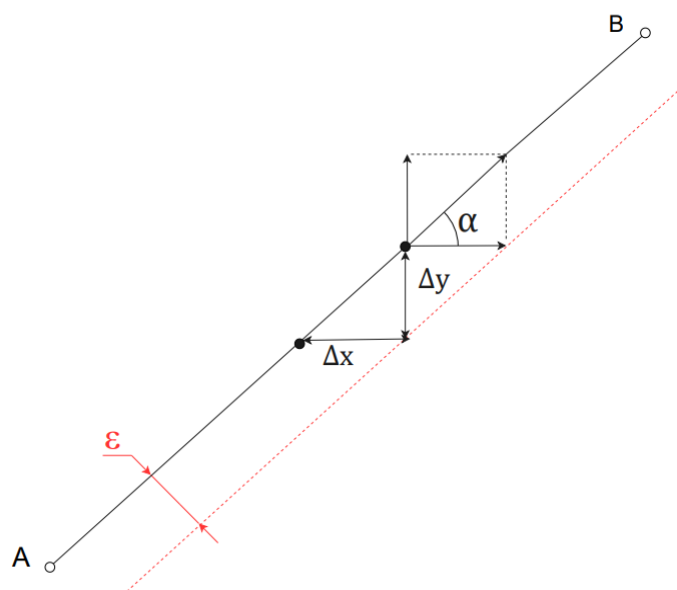
Sloučením těchto rovnic lze vyjádřit rovnice skuteční trajektorie.

$$y + \frac{v_y}{K_{vy}} - y_1 = \frac{v_y}{v_x} (x - x_1) + \frac{v_y}{K_{vx}} \quad (9)$$

Velikosti polohových odchylek ve směrech x a y se vektorově sečtou do směru rychlosti v_0 .

Polohová odchylka ε , zobrazená na obrázku 26 se vyjádří dle vztahu:

$$\varepsilon = \frac{v_0}{2} \sin 2\alpha \left| \frac{K_{vx} - K_{vy}}{K_{vx} \cdot K_{vy}} \right| \quad (10)$$



Obrázek 22: Odchylka při lineární interpolaci

V případě kruhové interpolace jsou vztahy pro souřadnice, rychlost a zrychlení následující:

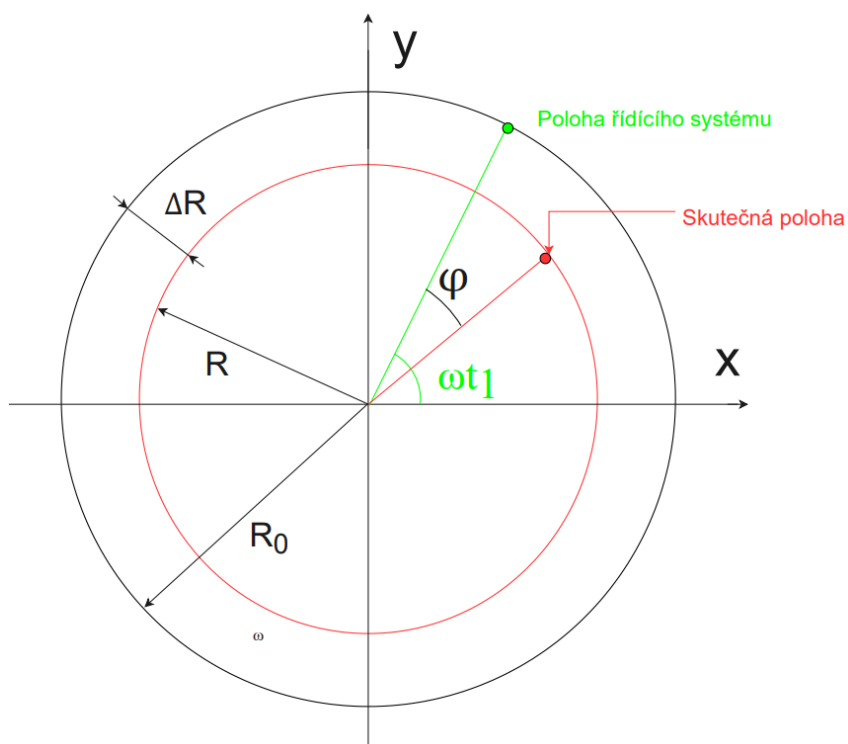
$$x = R_0 \cos \omega t \quad (11)$$

$$y = R_0 \sin \omega t \quad (12)$$

$$v_B = R_0 \omega \quad (13)$$

$$\ddot{x} = -R_0 \omega^2 \cos \omega t \quad (14)$$

$$\ddot{y} = -R_0 \omega^2 \sin \omega t \quad (15)$$



Obrázek 23: Odchylka kruhové interpolace

Souřadnice skutečné polohy se dají určit dle vztahů níže:

$$x_{skut} = R_0 \cos(\omega t_0 - \varphi) \quad (16)$$

$$y_{skut} = R_0 \sin(\omega t_0 - \varphi) \quad (17)$$

Odchylka ΔR lze vyjádřit dle rovnice:

$$\Delta R = -\frac{v_{AB}^2}{2R_0 K_v^2} \quad (18)$$

Příčemž:

$$v_{AB} = R_0 \omega \quad (19)$$

Obecně platí, že chyba ΔR se zmenšuje se zvyšující se hodnotou K_v a rostoucím poloměrem kružnice R_0 .

6. Retrofit 3D tiskárny Stratasys Dimension Elite/768

FDM 3D tiskárna Stratasys Dimension Elite/768 má velikost tiskového prostoru 203 x 203 x 305 mm. Konstrukční uspořádání pohybových os je kartézské, každá osa má jeden krokový motor, lineární pohyb os tak odpovídá změnám polohy tiskové hlavy.



Obrázek 28: Stratasys Dimension Elite/768 (20)

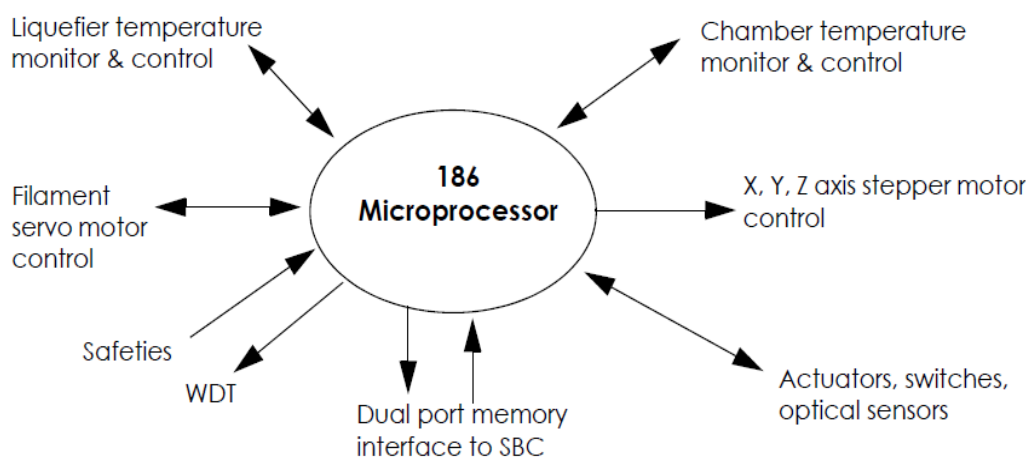
V tiskové hlavě jsou umístěny dva extrudéry, pod tiskovým prostorem pak dva zásobníky na špulky s filamentem. Pro posun filamentu od začátku bovdenu k motoru na extrudéru zajišťují samostatné kartáčové stejnosměrné motory. (23)

6.1. Řízení tiskárny

Procesor Intel 80186 s EPROM a DPRAM pamětí, na kterém běží operační systém Linux, poskytuje veškeré nízko úrovněvé ovládání a detekci stavů. Pracuje na frekvenci 16MHz.

Dual-port RAM

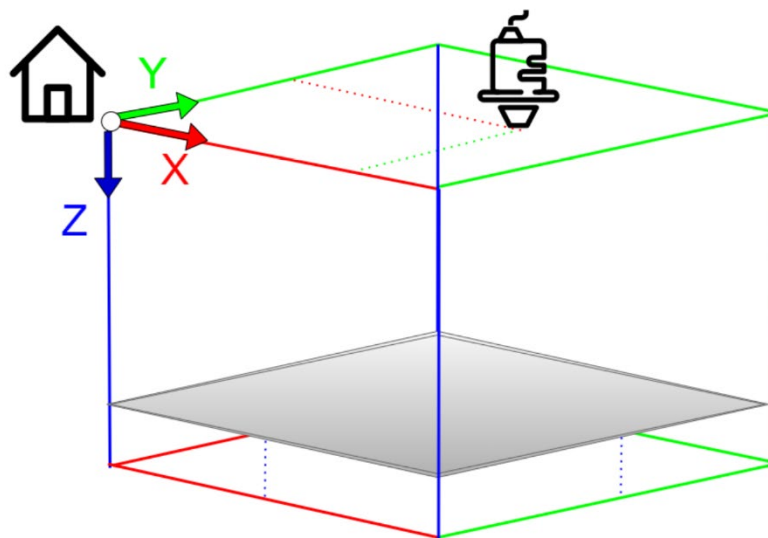
DPRAM je paměť, která umožňuje více čtení nebo zápisů současně, na rozdíl od jednoportových RAM. Tato paměť, umístěná na 16bitovém procesoru Intel 80186 (obrázek 29), poskytuje komunikační kanál se Single board počítačem (SBC). SBC poskytuje příkazy pro souřadnice, rychlosti. Procesor Intel 80186 poskytuje stavové informace o hardwaru do SBC.



Obrázek 29: Služby, které řídí procesor Intel 80186 (23)

Ovládání pohybových os X, Y, Z

Procesor Intel 80186 převezme souřadnicové a rychlostní vstupy z SBC a odešle je do PMD čipů pro řízení krokových motorů. PMD drivery používají výše uvedené mikrokrokování. Motory nemají zpětnou vazbu, pracují v otevřené regulační smyčce. Uspořádání os je znázorněno na obrázku 30.



Obrázek 30: Uspořádání os na tiskárně Stratasys

Ovládání motoru posunu filamentu

Procesor Intel 80186 bere informace o dataflow z SBC a odesílá je do driveru LM629M-6 pro řízení kartáčových DC motorů. Driver na základě dat požadavku z SBC a pulsů z enkodéru řídí motor filamentu. Na této tiskárně jsou dvě tiskové hlavy, každá má vlastní driver LM629M-6. Vzhledem ke zpětné vazbě se jedná o uzavřenou regulační smyčku. Stále víme, kde motor je. Pro kvalitní 3D tisk metodou FDM je zásadní, aby byl průtok nataveného filamentu stále stejný.

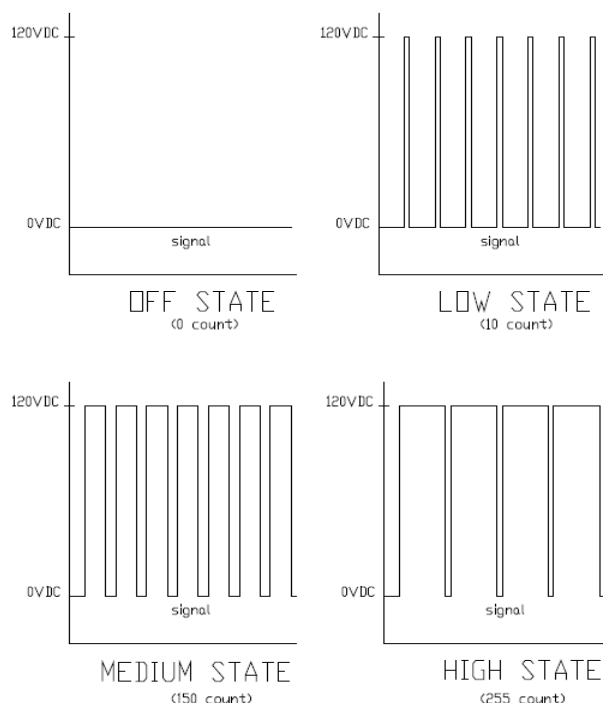
Regulace teploty tiskové komory

Termočlánek umístěný v prostoru tiskové komory je připojen k řídicí desce, kde je malý proud z termočlátku zesílen a převeden z analogového signálu na digitální pomocí A/D převodníku. Procesor na základě vyhodnocené teploty ovládá topná tělesa, aby byla v tiskové komoře udržována teplota od 70°C do 75°C. Při temperování tiskové komory je vždy sepnut vnitřní ventilátor.

Regulace teploty trysky

Zde je teplotní senzor, též termočlánek umístěn v těle tiskové hlavy blízko trysky filamentu. Teplotní regulace probíhá stejně, jako výše uvedená regulace teploty tiskové komory. Teplota, na kterou se tryska vyhřívá se pohybuje od 180°C do 270°C.

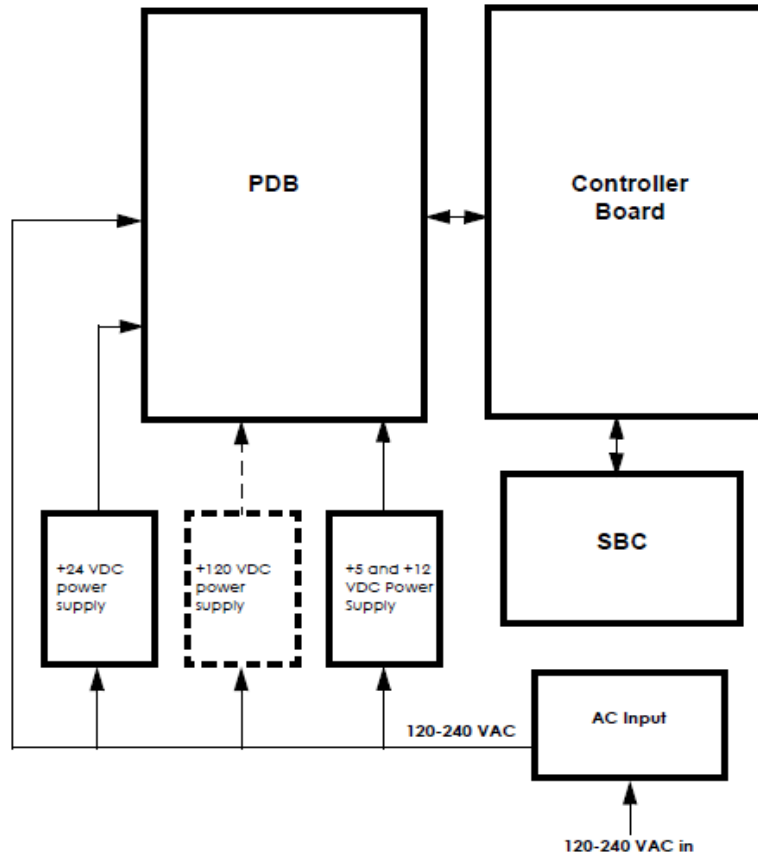
Pro řízení výkonu topných těles je použita pulsní šířková modulace (PWM) – obrázek 31. V tomto případě umožňuje 1000 pulsů za sekundu při napájecím napětí 120VDC.



Obrázek 31: Příklad pulsní šířkové modulace (23)

Rozvaděčová skříň

V zadní části těla tiskárny Stratasys je místo pro řídicí a napájecí komponenty. Základní prvky jsou zdroj hlavního napětí pro celou tiskárnu – AC Input, jednotlivé napájecí zdroje pro 5 VDC, 12 VDC, 24 VDC a 120 VDC. Dále PDB – Power Distribution Board, která zajistí správné napájení příslušných komponent; Cotroller Board zahrnuje spínací prvky a SBC Single Board Computer, který řídí procesy.



Obrázek 32: Schéma řídicích a napájecích prvků (23)

Komponenty 3D tiskárny dle napájecího napětí (obrázek 32):

Motory osy – 24VDC

Motory filament – 12VDC

Koncové snímače, zámky, osvětlení, ventilátory – 24VDC

Temperování trysky a komory – 120VAC

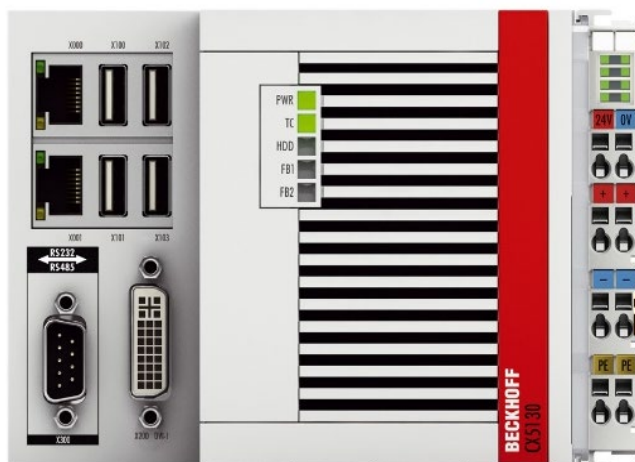
6.2. Řídicí systém Beckhoff

Retrofit řídicího systému 3D tiskárny Stratasys spočívá v nahrazení stávajícího řídicího systému. Jako náhrada byl zvolen průmyslový počítač (IPC) CX5130 od společnosti Beckhoff (obrázek 33).

Zařízení IPC jsou založena na PLC systémech, hlavní rozdíl spočívá v tom, že na IPC mohou být instalovány operační systémy Windows a Linux, dále nabízí více paměti a větší flexibilitu.

Použité IPC CX5130 má 4 jádrový procesor Intel Atom® s taktovací frekvencí 1,75 GHz. Možnost komunikace je přes dvě nezávislá ethernetová rozhraní s možností gigabitového připojení, čtyři USB 2.0 a rozhraní DVI-I.

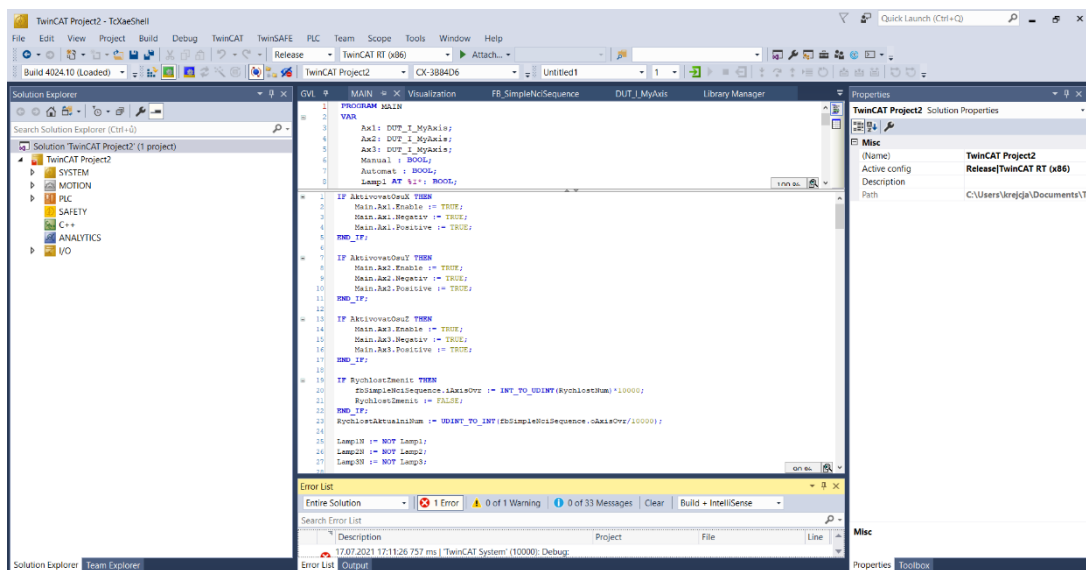
K IPC lze připojit různé moduly (karty), které umožní připojení na jednotlivé periferie.



Obrázek 33: IPC Beckhoff CX5130 (33)

TwinCat

TwinCAT PLC Control je kompletní vývojové prostředí řídicí systémy Beckhoff. Poskytuje jednoduchý přístup k výkonnému jazyku IEC. Použití editorů a funkcí ladění je založeno na standartních prostředích vývojového programu programovacího jazyku IEC 61131-3. Veškerá funkcionalita TwinCat PLC Control je zároveň obsažena v instalačním modulu, nainstalovaný do softwarového prostředí. Podoba programového prostředí je na obrázku 34.



Obrázek 34: Hlavní okno programového rozhraní TwinCat

IEC 61131-3

Norma IEC 61131-1: 2003 se vztahuje na programovatelné řadiče (PLC) a jejich přidružené periferní zařízení, jako jsou programovací a ladicí nástroje (PADT), rozhraní člověk-stroj (HMI) atd., jejichž účelem je ovládání strojů a průmyslových procesů. Poskytuje definice pojmů použitých v této normě. Identifikuje hlavní funkční charakteristiky programovatelných řídicích systémů.

Norma se skládá z následujících částí:

1. Obecné informace
2. Požadavky na zařízení a zkoušky

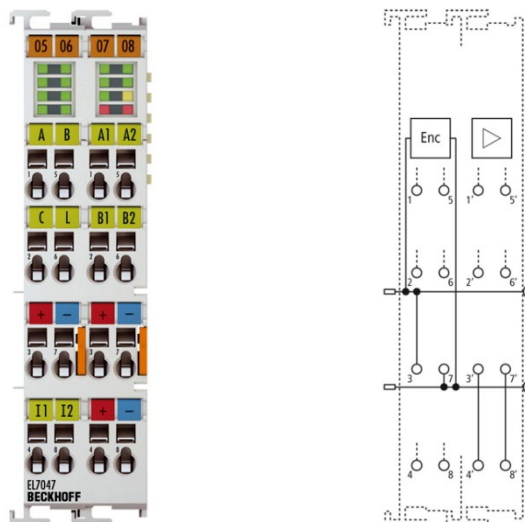
3. Programovací jazyky
4. Pokyny pro uživatele
5. Komunikace
6. Funkční bezpečnost
7. Programování fuzzy řízení
8. Pokyny pro aplikaci a implementaci
9. Rozhraní digitální komunikace pro senzory a akční členy

IEC 61131-3 je třetí částí rodiny IEC 61131, jeho nejnovější vydání obsahuje několik významných rozšíření předchozí edice jako např. nové datové typy, jmenné prostory a objektově orientované funkce atd. IEC 61131-3 je široce přijímaný programovací standard v automatizačním průmyslu v současnosti.

Téměř všichni výrobci PLC programují svůj systém podle této normy. IEC 61131-3 standard definuje nejen programovací jazyky, které mají být aplikovány v programování PLC, ale také poskytuje společné datové typy, které lze mezi těmito jazyky sdílet. IEC 61131-3 umožňuje programování pomocí Ladder diagramu (LD) a Funkčního blokového schéma (FBD) a dvou textových reprezentací - Seznamu instrukcí (IL) a Strukturovaného textu (ST) a jednoho sekvenčního Funkčního grafu (SFC), jehož prvky lze použít ve spojení s jedním z těchto jazyků. Každý z programovacích jazyků může být použit samostatně. (3)

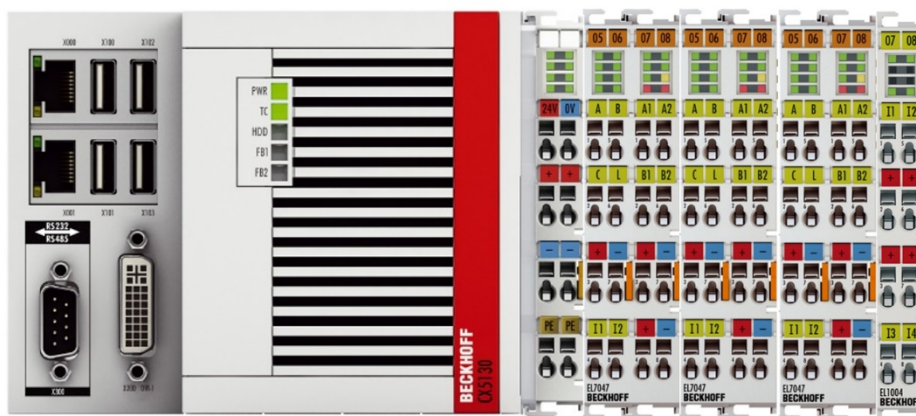
6.3. Propojení tiskárny Stratasys a řídicího systému Beckhoff

Karta EL7047 EtherCAT (obrázek 34) je určena k přímému připojení krokových motorů v proudovém rozsahu do 5 A, a napájecím napětí do 48 V. Prostřednictvím parametrizace v programu TwinCat lze tento modul přizpůsobit motoru a požadavkům aplikace. Motor se ovládá mikrokrokováním s možností dělení základního kroku na 64 inkrementů. Tato karta obsahuje i vstupy pro signály z enkodéru. V případě řízení tiskárny Stratasys, se se zpětnou vazbou nepočítá.



Obrázek 34: Driver krokových motorů EL7047 (34)

Pro ovládání krokových motorů na osách X, Y a Z potřebujeme 3 karty EL7047, zapojené do IPC CX5130. Dále je potřeba použít katu digitálních vstupů pro připojení koncových spínačů. Pro tento projekt je použita karta EL1004, která disponuje čtyřmi spínacími digitálními vstupy. Finální sestava je na obrázku 35.



Obrázek 35: Sestava IPC s kartami pro řízení os 3D tiskárny (33,34)

Koncové spínače

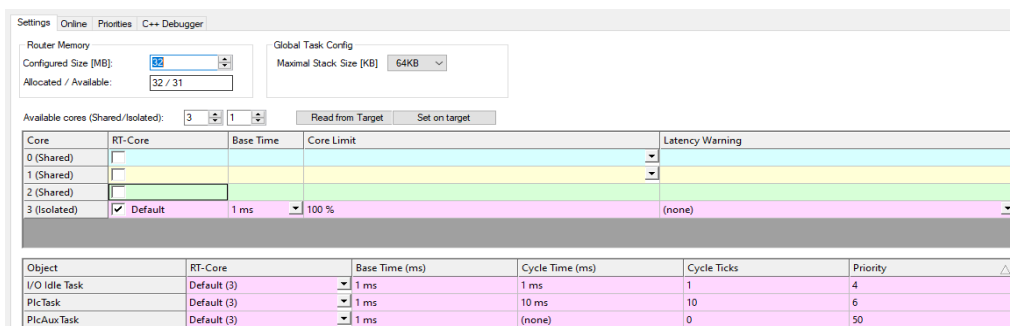
Pro určení referenčních poloh všech pohybových os je třeba použít referenční spínače. K tomuto účelu se používají indukční snímače, jazýčkové mikrospínače, magnetické, či optické snímače. V případě tiskárny Stratasys jsou použity optické snímače v podobě vidličky ve verzi vysílač – přijímač (obrázek 36). Při protnutí optického paprsku, je uzavřen obvod napájený z karty pro digitální vstupy.



Obrázek 36: Koncový spínač Hamlin 59085 (35)

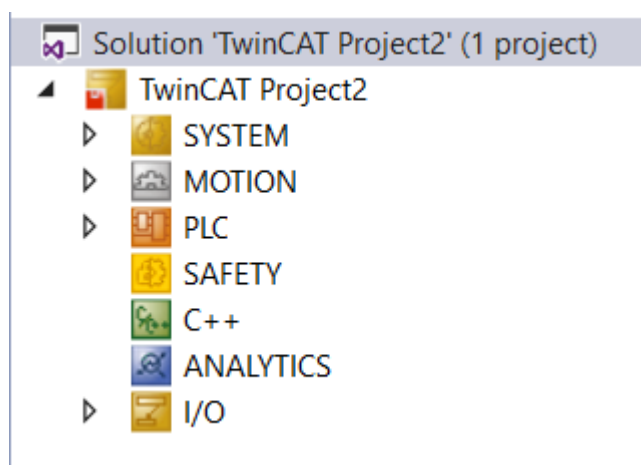
7. Tvorba programu

Před začátkem programování zařízení Beckhoff v prostředí TwinCat je třeba pro spolehlivé fungování alokovat jedno jádro procesoru počítače pouze pro aplikaci TwinCat. Windows tak uvidí pouze tři jádra pro svoje fungování ze čtyř dostupných, což je vidět na obrázku 37.



Obrázek 37: Alokační jádra procesoru pro TwinCat

Při založení nového PLC projektu dojde k vytvoření sedmi skupin, SYSTEM, MOTION, PLC, SAFETY, C++, ANALYTICS, I/O. Pro tvorbu programu je třeba skupiny SYSTEM – správa licencí, MOTION – konfigurace pohonů, PLC – program s proměnnými a I/O – hardwarová konfigurace, správa parametrů připojených karet.



Obrázek 38: Skupiny nového projektu

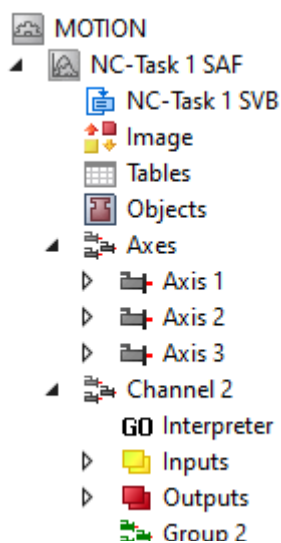
System

Skupina *System* (obrázek 38) slouží primárně ke správě licencí. Stažení vývojového prostředí TwinCat je zdarma. Dále se uvolňují licence pro jednotlivé moduly TwinCat, které v sobě obsahují specifické funkce a připravené funkční bloky – FBC. Kromě základního modulu pro PLC – TC1200 je třeba aktivovat licenci modulu NC I – TF5100.

TwinCAT NCI je zkratka pro „numerickou regulační interpolaci“ a je NC systémem pro interpolovaný pohyb, který má vlastní interpreter i regulátor polohy.




Pro účely programování a testování není třeba za licence platit. Je zde možnost týdenní trial verze, která se znovu aktivuje automaticky generovaným kódem.

Motion



Obrázek 39: Skupina Motion

Ve skupině *Motion* (obrázek 39) se vytvoří NC úloha, do které se přiřadí pohybové osy, v tomto případě *Axis 1 – Axis 3*. Tyto osy se přiřadí jak k fyzickým kartám připojených do PLC, tak na stranu softwaru – do proměnných GVL.ax1 – GVL.ax3 (obrázek 40). Osy se přidávají do tzv. kanálu – *Channel* a vytvoří kinematickou skupinu – *Group 2*.

Number	Axes	Link to I/O	Link to PLC
 1 (Id 1)	Axis 1	Term 5 (EL7047)	GVL.ax1 (Untitled1 Instance)
 2 (Id 2)	Axis 2	Term 6 (EL7047)	GVL.ax2 (Untitled1 Instance)
 3 (Id 3)	Axis 3	Term 7 (EL7047)	GVL.ax3 (Untitled1 Instance)

Obrázek 40: Přřazení os

PLC

Ve skupině PLC se vytváří samotný řídicí program. Záložka Refernces, obsahuje proměnné a funkce spjaté s licencemi, které byly aktivovány. Dále jsou tu skupiny DUTs, GVLs, POUs a VISUs.

DUT (Data Unit Types) je speciální objekt, který umožňuje definovat strukturu. Struktura funguje stejně jako standardní datový typ (například Bool nebo Real) a umožňuje definovat proměnné jako typ struktury. V tomto projektu bylo DUT použito pro vytvoření struktury pro spouštění osy – obrázek 41.

```

1  TYPE DUT_I_MyAxis :
2  STRUCT
3      Enable: BOOL;
4      Positive: BOOL;
5      Negativ: BOOL;
6      JogForward: BOOL;
7      JogBackWard: BOOL;
8  END_STRUCT
9  END_TYPE
10

```

Obrázek 41: DUT

GVL – Global Variable List neboli list globálních proměnných je výčet proměnných a konstant, které jsou rozpoznávány v rámci celého projektu.

POU se nazývá Program Organisation Unit a používá se k navrhování a provádění uživatelských kódů pro PLC.

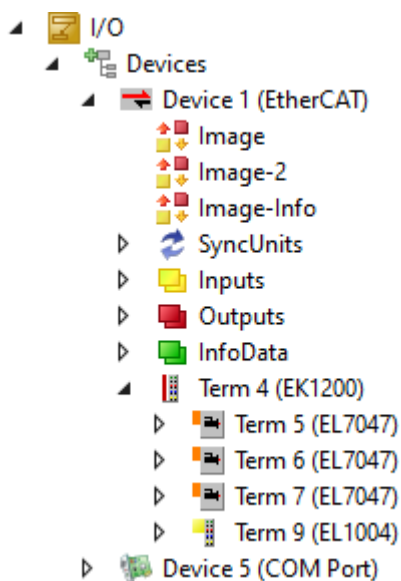
POU jsou tři druhy:

1. program – aplikační kód kombinující vstupy, výstupy, funkce a funkční bloky,
2. funkce – opakovaně použitelný uživatelský kód, který má pouze jeden výstup s návratovou hodnotou,
3. funkční blok – opakovaně použitelný uživatelský kód s více než jedním možným výstupem, vyžaduje instanci pro uložení interních hodnot.

Pro psaní programu je několik možností implementačního jazyka, nejpoužívanější je FBD – Funkční blokové schéma, což je grafické znázornění programu složené z jednotlivých bloků s vnitřní logikou navázanou na vstupy a výstupy. LD – Ladder diagram. Ladder – žebřík v názvu připomíná grafické zobrazení, kdy jednotlivé příčky značí řídicí obvody. Nejpoužívanějším a nejkompexnější interpretační jazyk je ST – Strukturovaný text, který má kořeny ve známých jazycích Ada, Pascal a C. Obsahuje všechny podstatné prvky moderního programovacího jazyka, včetně větvení (IF-THEN-ELSE a CASE OF) a iterační smyčky (FOR, WHILE a REPEAT). Dané prvky mohou být vnořovány. Tento jazyk je vynikajícím nástrojem pro definování komplexních funkčních bloků, které pak mohou být použity i v jiném programovacím jazyku. Stukturovaný text je nejkompexnější možností a byl zvolen pro psaní programu pro řízení 3D tiskárny Stratasys.

I/O

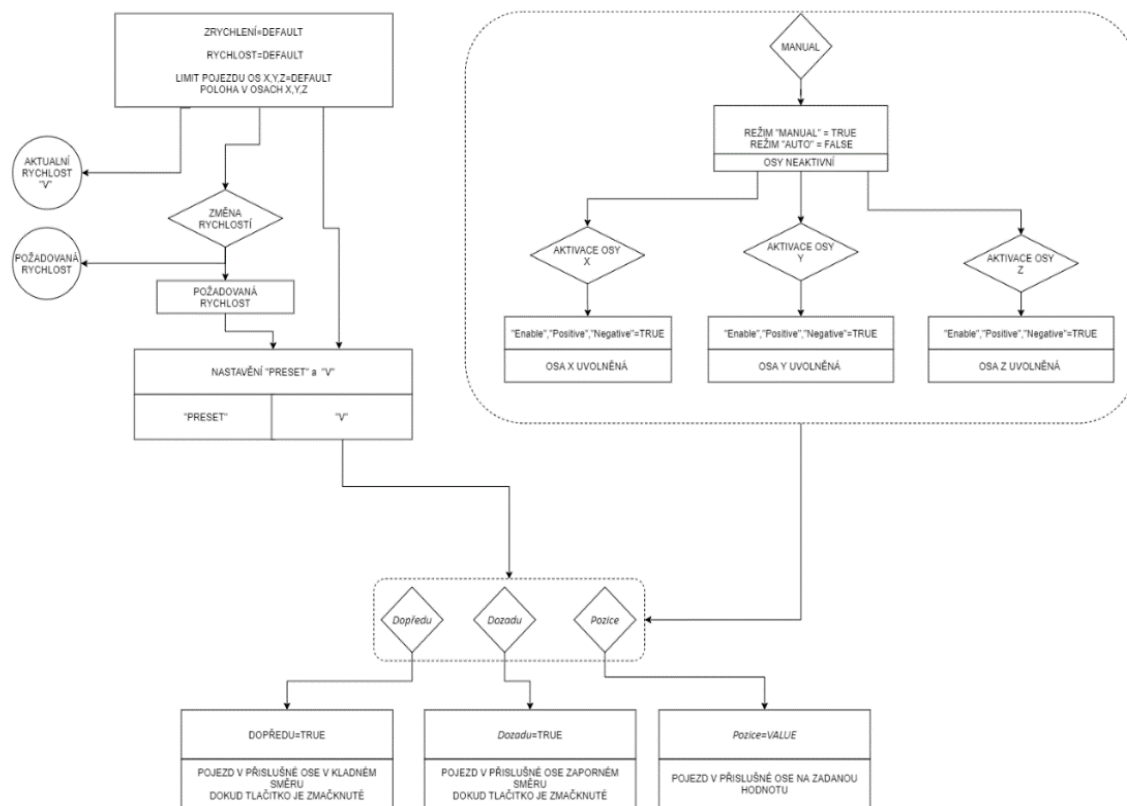
Modul I/O obsahuje hardwarovou konfiguraci připojených zařízení, z obrázku 42 lze vyčíst připojení třech modulů EL7047 pro řízení krokových motorů a jednoho modulu EL1004 do coupleru, který je součástí IPC CX5130. Nastavení parametrů pro drivery se provádí zde v záložce CoE, což je podrobněji popsáno v kapitole Nastavení krokových motorů 3D tiskárny.



Obrázek 42: Modul I/O

7.1. Popis funkce programu

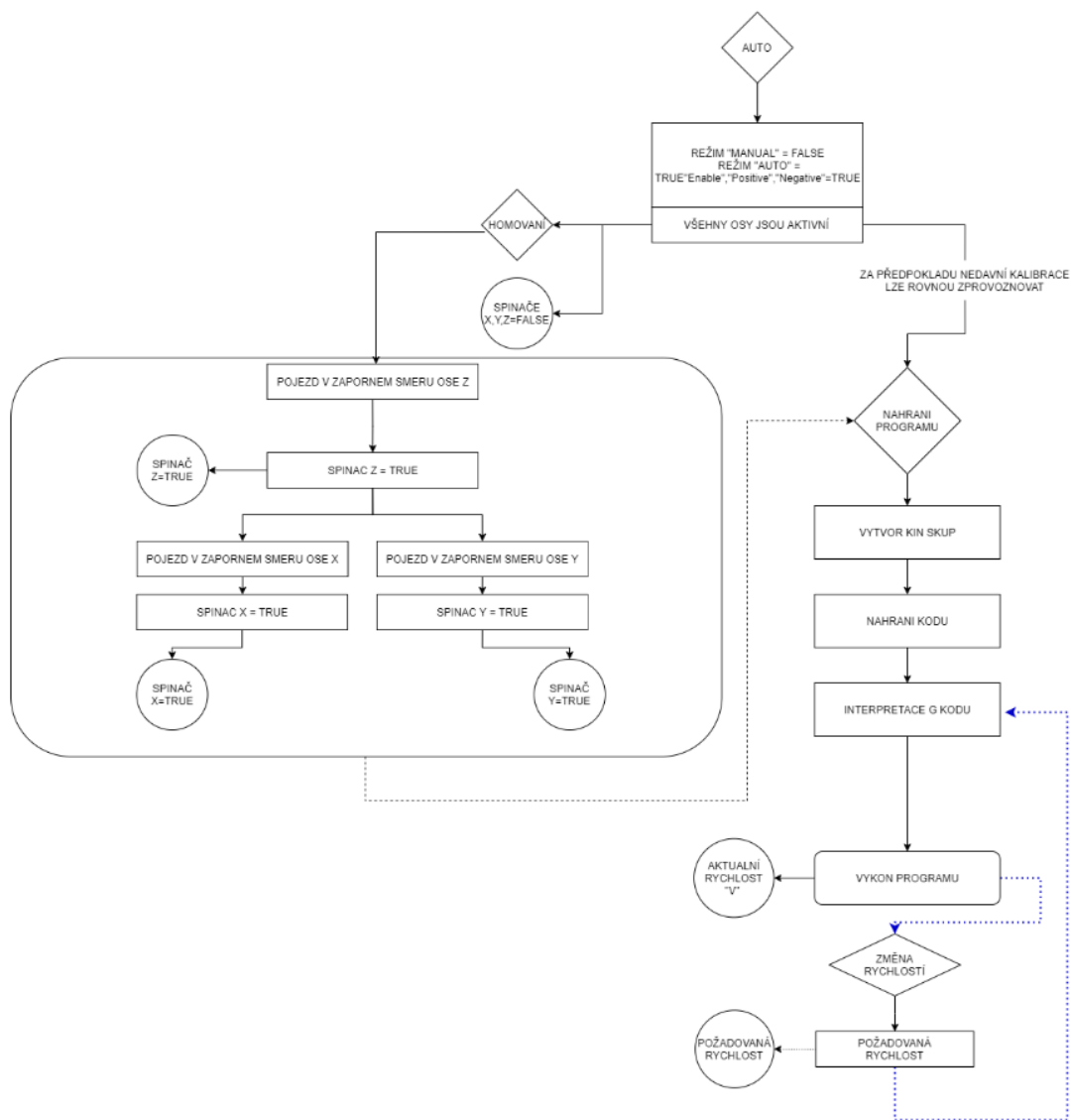
Program má dva základní režimy, *Manual* a *Automat*. V režimu *Manual* lze ovládat jednotlivé osy 3D tiskárny. Pro uvolnění jednotlivých motorů je třeba propsat hodnotu *TRUE* do funkcí *Enable*, *Positive*, *Negative*, které jsou vázané na proměnné v programu. Po změně proměnných lze s osou pohybovat, stiskem tlačítek *Dopředu* a *Dozadu*. Další možností je najetí na konkrétní pozici. Do pole pod tlačítkem *Pozice* se zadá hodnota v milimetrech. Polohování je absolutní. Pro větší uživatelský komfort bylo pod okno pro zadání *Pozice* přidáno tlačítko *Aktivovat osu X*. Stiskem se aktivují proměnné *Enable*, *Positive* i *Negative*. Tyto možnosti ovládání se opakují i pro osu Y a Z. Rychlost pohybových os lze měnit v dolní pravé části uživatelského rozhraní. Do horního pole se zadá požadovaná hodnota a po stisku tlačítka *Změna rychlosti* dojde k propsání hodnoty do okna spodního. Stavový diagram režimu *Manual* je na obrázku 43.



Obrázek 43: Diagram manuálního režimu

Po stisku tlačítka *Automat* se aktivují všechny 3 osy. Jednotlivé osy nemají přímou ani nepřímou zpětnou vazbu je tak třeba před začátkem každého výrobního procesu určit referenční polohy os pomocí koncových spínačů. Po stisku tlačítka *Homování* se začne osa Z pohybovat směrem nahoru, dokud nedojde k sepnutí koncového spínače osy Z. Následně se dají do pohybu osy X a Y, též dokud nedojde k sepnutí koncových spínačů.

Stiskem tlačítka *Zpracuj kód* se spustí stavový automat pro načítání NC programu v G-kódu, který se zpracovává interpretem a začíná interpolovaný pohyb. Změna rychlosti lze aplikovat též na běžící program. Tlačítko *Reset* maže chybové hlášky a resetuje jednotlivé motory. Stavový diagram popsáního děje je na obrázku 44.



Obrázek 44: Diagram automatického režimu

7.2. Řízení os 3D tiskáren

Pro obráběcí stroje se běžně používá kaskádní regulace pohonů, která je založena na navzájem navazujících regulačních obvodech se zpětnou vazbou. Hlavním důvodem je potřeba dosažení vysoké přesnosti polohování u strojů, kde dochází ke vzniku velkých dynamických sil. Neustálá změna rychlosti, pasivní odpory, zatěžující síly od obrábění ovlivňují výrobní proces a tím i kvalitu povrchu či rozměrové tolerance výrobku. Zcela jiná

je situace u 3D tiskáren. Dochází zde pouze ke vzniku zanedbatelných pasivních odporů. Standardní rychlosti posuvu tiskové hlavy u FDM technologie se pohybuje okolo 50 mm/s. Vyrábějí se i tiskárny, kde výrobce deklaruje maximální rychlost 500 mm/s, jedná se ovšem většinou o konstrukční provedení typu SCARA. U 3D tiskáren kartézské konstrukce je převedení rotačního pohybu hřídele krokového motoru realizováno pomocí ozubeného řemenového kolečka o průměru 16 mm a ozubeného řemenu. Při rychlosti posuvu 50 mm/s to dle níže uvedeného vztahu odpovídá jedné otáčce hřídele za sekundu.

$$n = \frac{v}{\pi \cdot d} = \frac{200}{\pi \cdot 16} = 0,995 \cong 1 \text{ ot/s} \quad (20)$$

Osa Z vykonává pohyb pouze při změně tiskové vrstvy a při najíždění do referenční polohy. Tloušťka tiskové vrstvy se pohybuje od 0,05 do 0,4 mm. Více než na rychlost je kladen nárok na rozlišení osy. Převod rotačního pohybu na lineární je provedeno přes trapézový či kuličkový šroub.

Spolu s trhem vyžadovanou nízkou pořizovací cenou a menším požadavkem na výslednou kvalitu a přesnost výrobků je dostačující zvolení jednoduššího typu regulace ve formě otevřené smyčky. Tato forma řízení je dána řídicím systémem a řízenou sestavou, kde řídicí systém generuje signál za předpokladu známého jednoduchého chování řízené sestavy.

Řízení krokových motorů

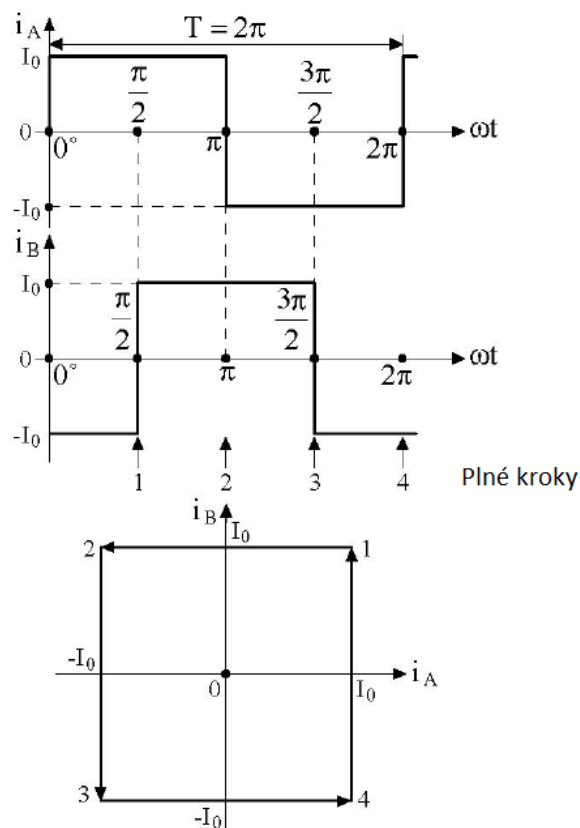
Krokové motory byly původně provozovány s velmi jednoduchými koncovými stupni, které dokázaly pouze přepínat napětí fází motoru samostatně v současné době probíhá řízení proudu pomocí PWM. Zpočátku byly motorové fáze postupně ovládány jednotlivě. Přepínání posloupnosti v kladném směru otáčení odpovídá spínací posloupnosti (+ A, + B, -A, -B). Sekvenční přepínání vede v tomto režimu k poměrně nepravidelnému provozu. Aby bylo možné provést operaci používá se dnes metoda mikrokrokování.

Mikrokrokování

Pro mnoho aplikací je příhodné jemnější krokování než nám umožňuje konstrukce motoru spolu s uvedenými metodami řízení. Standardně bývá jeden krok od 0,36 do 15 stupňů.

Další rozdělení tohoto úhlu umožňuje mikrokrokování, při kterém je možno každý krok rozdělit až na 128 mikrokroků. Při klasickém buzení krokového motoru prochází jednotlivými fázemi proudy stejné velikosti. Při mikrokrokování se budicí proud na cívkách postupně mění, aby bylo možno nastavit polohu mezi dvěma kroky. Obvykle je budicí proud podobný sinusovému průběhu. Přivedením proudů na obě fáze motoru vznikne momentový fázor, který je úměrný vektorovému součtu obou proudů. Poloha motoru je tedy určena úhlem fázorového momentu. Běžný způsob ovládání úhlu fázorového momentu je aplikováním na dvě periodické křivky fáze motoru posunutě o 90 stupňů. Rovnice pro určení jednotlivých proudů jsou:

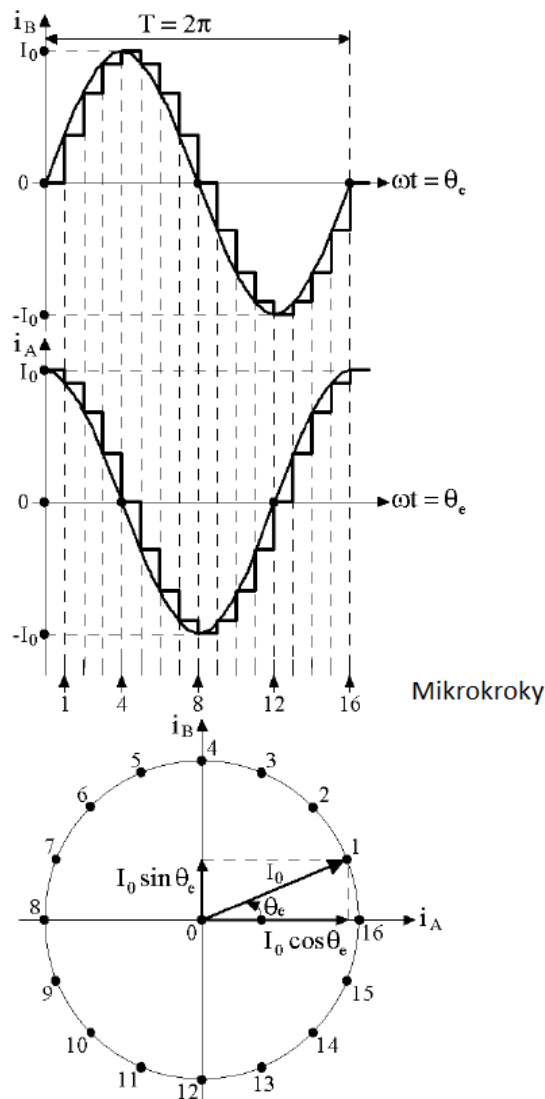
$$i_A = I_0 \cdot \cos\theta_e \quad (21)$$



Obrázek 45: Průběhy proudů pro plné kroky (31)

$$i_A = I_0 \cdot \sin \theta_e \quad (22)$$

Pomocí této metody řízení můžeme dosáhnout teoreticky jakéhokoli rozlišení. Vzhledem k tomu, že průběhy budicího proudu jsou namísto čtvercových sinusové, jsou oscilace mezi kroky eliminovány. Dochází ke zlepšení funkčnosti za nízkých otáček a snížení rezonančních problémů. Příklad požadovaných fázových proudů pro celý krok a mikrokroky je na obrázku 45 a 46. (31)



Obrázek 46: Průběh proudů při mikrokrokování (31)

Nastavení polohování

Správné fungování pohybových os je závislé na nastavení krokových motorů. Každému driveru náleží množství parametrů, které jsou, až na převodový poměr u osy Z, nastaveny stejně pro všechny pohony. Ve skupině parametrů 8010:0 jsou parametry související s konstrukcí motoru.

Motory použité v tiskárně Stratasys mají úhel jednoho kroku $1,8^\circ$. Tento parametr je třeba nastavit do parametrů driveru jako počet kroků na jednu otáčku:

$$\text{Počet kroků na otáčku} = \frac{360^\circ}{1,8^\circ} = 200 \text{ kroků} \quad (23)$$

Dalším parametrem je *Scaling factor (SF)* kde je výsledná hodnota závislá na počtu mikrokroků:

$$SF = \frac{360^\circ}{\text{Počet kroků na otáčku} \cdot \text{počet mikrokroků}} = \frac{360^\circ}{200 \cdot 64} = 0,028125^\circ \quad (24)$$

Pro pohyb tiskové hlavy je nutné převést rotační pohyb krokových motorů na lineární posuv. Požadovaný poměr můžeme získat buď přepočtem převodu ozubeného řemenového kolečka, či stoupání trapézového šroubu u osy Z, případně přesným měřením s porovnáním požadovaných a skutečných hodnot. Jako relevantnější byla použita druhá metoda. Pro vyhodnocení vzdálenosti byl použit senzor Sick OD5000-C85T20 (Obrázek 47). Senzor pro měření využívá laserového paprsku, který se odrazí od měřeného předmětu do CMOS senzoru. S rostoucí vzdáleností předmětu do senzoru dochází ke zmenšení úhlu mezi vyslaným a odraženým paprskem. Poloha dopadu na CMOS senzor je následně přepočtena na reálnou vzdálenost. Opakovatelnost senzoru je $0,1 \mu\text{m}$. Linearita měření $7,2 \mu\text{m}$.



Obrázek 47: Triangulační laserový senzor OD5000-C85T20 (37)

Vlastnosti senzoru jsou tedy více než dostačující pro relevantní měření. Měřené hodnoty lze sledovat a ukládat v parametrizačním softwaru – Obrázek 48.



Obrázek 48: Průběh měření pohybu os

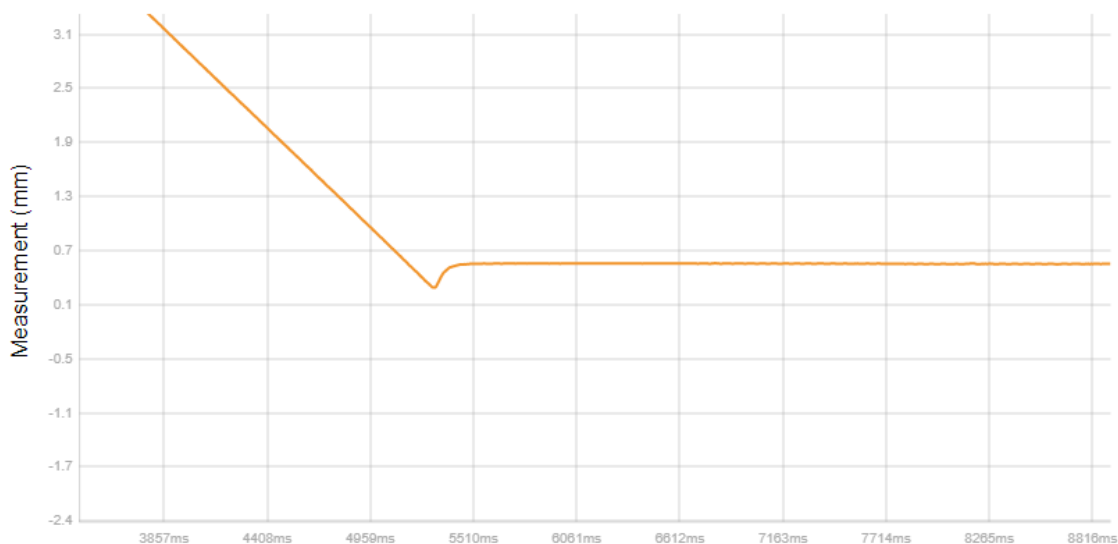
Pro každou osu bylo provedeno 5 měření na základě kterého byl vypočítán převodový poměr, který se zadá jako parametr *Scaling Factor Denominator* v tabulce na obrázku 49.

Pro osy X a Y je to hodnota 9,0548, pro osu Z 44,846.

Parameter	Offline Value	Online Value	T.	Unit
Encoder Evaluation:				
Invert Encoder Counting Direction	FALSE	FALSE	B	
Scaling Factor Numerator	0.028125	0.028125	F	mm/INC
Scaling Factor Denominator (default: 1.0)	9.054	9.054	F	
Position Bias	0.0	0.0	F	mm
Modulo Factor (e.g. 360.0°)	360.0	360.0	F	mm
Tolerance Window for Modulo Start	0.0	0.0	F	mm
Encoder Mask (maximum encoder value)	0x0000FFFF	0x0000FFFF	D	
Evaluation Direction (log. count direction)	'POS+NEG'	'POS+NEG'	E	

Obrázek 49: Nastavení parametrů 1

Velká přesnost senzoru umožňuje dobré ladění překmitů regulační konstantou K_V . Znatelný polohová regulační odchylka je zřejmá z obrázku 50.



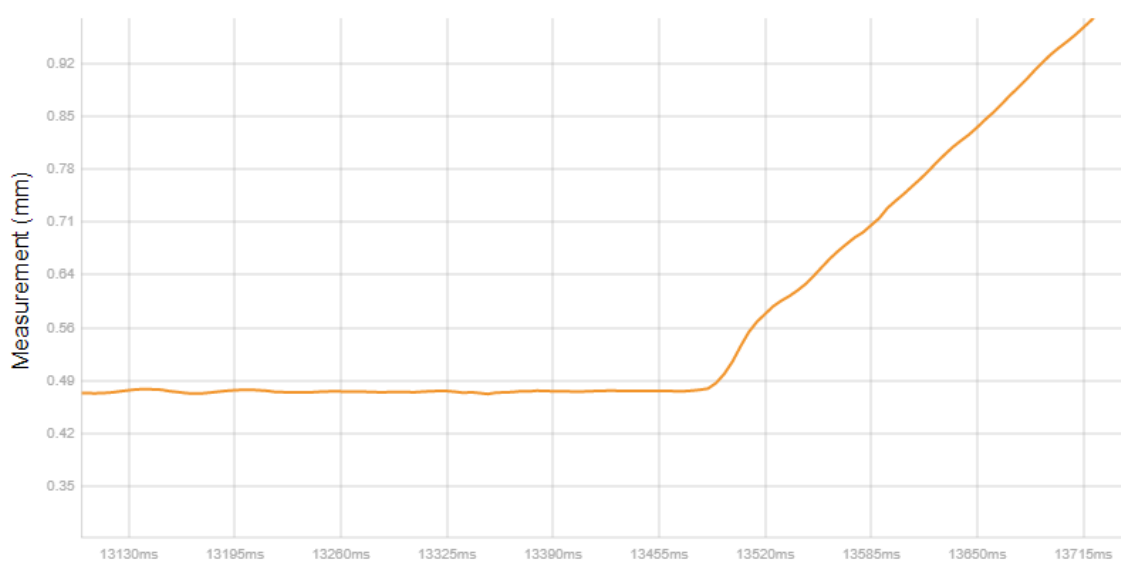
Obrázek 50: Znatelný překmit při dojetí na pozici

Na obrázku 49 je náhled parametrizačního okna s parametry pro skupiny 8010:0, 8011:0, 8012:0, které vycházejí z konstrukce motoru a elektrických parametrů – obrázek 51.

8010:0	STM Motor Settings Ch.1	RW	> 17 <	
8010:01	Maximal current	RW	0x0258 (600)	mA
8010:02	Reduced current	RW	0x012C (300)	mA
8010:03	Nominal voltage	RW	0x01A6 (422)	0,01 V
8010:04	Motor coil resistance	RW	0x02A8 (680)	0,01 Ohm
8010:05	Motor EMF	RW	0x0000 (0)	mV/(rad...
8010:06	Motor fullsteps	RW	0x00C8 (200)	
8010:07	Encoder increments (4-fold)	RW	0x1000 (4096)	
8010:09	Start velocity	RW	0x0000 (0)	
8010:...	Motor coil inductance	RW	0x01F4 (500)	0,01 mH
8010:10	Drive on delay time	RW	0x0064 (100)	ms
8010:11	Drive off delay time	RW	0x0096 (150)	ms
8011:0	STM Controller Settings Ch.1	RW	> 2 <	
8011:01	Kp factor (curr.)	RW	0x0014 (20)	
8011:02	Ki factor (curr.)	RW	0x0014 (20)	
8012:0	STM Features Ch.1	RW	> 58 <	
8012:01	Operation mode	RW	Automatic (0)	
8012:05	Speed range	RW	2000 Fullsteps/sec (1)	
8012:08	Feedback type	RW	Internal counter (1)	

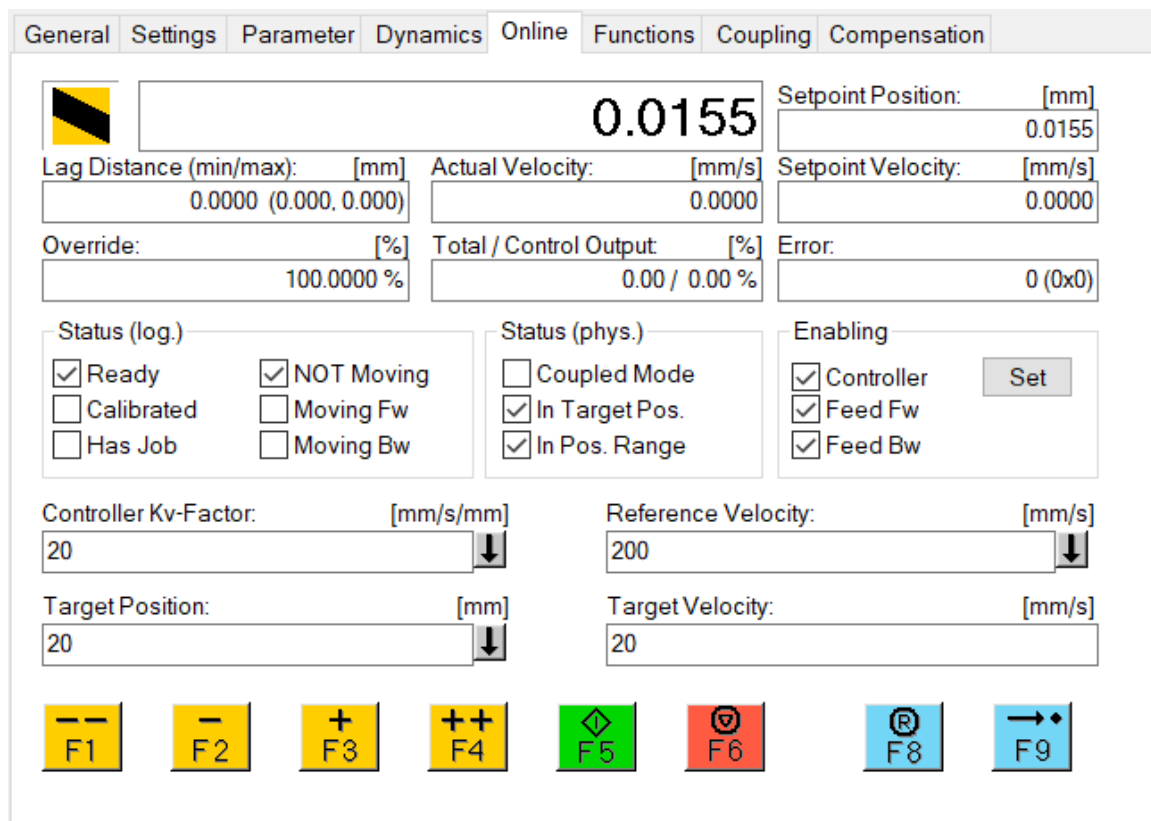
Obrázek 51: Nastavení parametrů 2

Na obrázku 52 je znázorněn časový průběh pozice osy po úpravě parametrů.



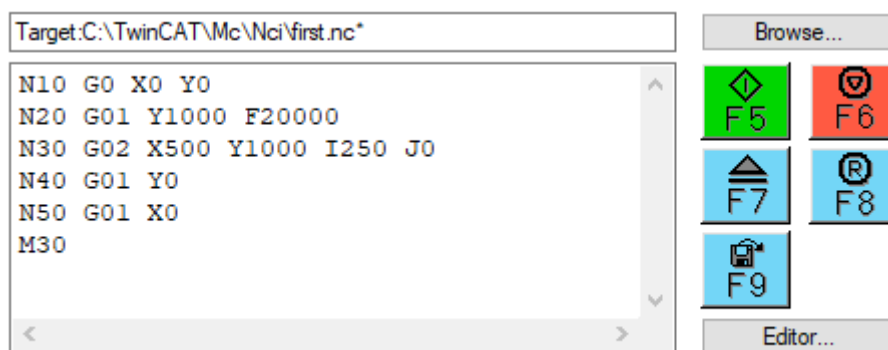
Obrázek 52: Časový průběh naladěného pohonu

Ovládání jednotlivých os je možné v prostředí TwinCat, v záložce Online (obrázek 53), což je velmi praktické pro rychlé ladění motorů a zadávání požadovaných pozic a rychlostí.



Obrázek 53: Prostředí pro ovládání pohonů

Po přidání os do společné skupiny a naladění krokových motorů je možné nahrávat NC programy uložené v paměti IPC (případně USB disk, použité IPC má 4 USB porty),



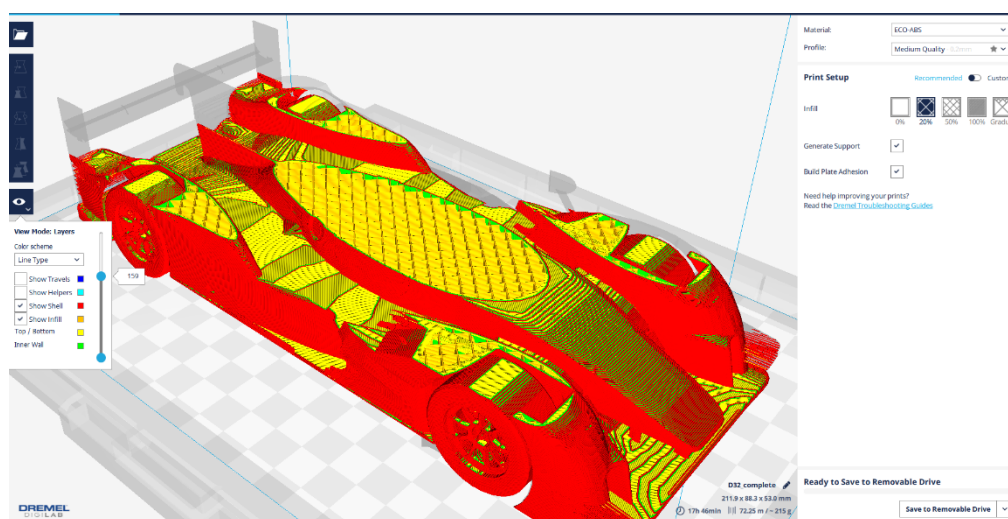
Obrázek 54: Ovládací okno s nahaným NC programem

provádět úpravy, spouštět program či resetovat chyby, viz obrázek 54. Zároveň je možné sledovat aktuální polohu jednotlivých os.

7.3. NC program

Při standardním procesu vzniku součásti obráběcími metodami je vytvořen výkres a CAD model, následně se v CAM softwaru, do kterého se CAD model nahraje, nastaví technologické operace. Poté se vygenerují CL data (cutter location), které zpracuje postprocessor. Postprocessor převádí CL data na data srozumitelná dané kombinaci stroj + řídicí systém, jedná se o určitý typ překladače. Tímto vznikne NC program v G-kódu. Pomocí interpretu řídicího systému jsou generovány pokyny pro jednotlivé osy.

V případě 3D tisku se místo CAM softwaru používá tzv. Slicer, což je software, který vložený 3D model rozřeže na jednotlivé vrstvy ve směru osy Z. Ve Sliceru se nastavují všechny parametry spojené s 3D tiskem. Základním parametrem je tloušťka jedné vrstvy. Tu je třeba vhodně zvolit podle požadavků na přesnost, kvalitu povrchu a přiměřeného času výroby. U menších tiskáren se nejčastěji používá 0,2 mm. Pokud se tedy bude tisknout například díl vysoký 200 mm, dojde k rozdělení na 1000 vrstev. Dále se ve sliceru volí rychlost tisku, průměr trysky filamentu, teplota trysky a tiskové podložky, či celé tiskové komory, počet perimetrů pro stěny dílce, procentuální množství výplně, její geometrická podoba a nastavení tiskových podpěr. Na základě těchto parametrů je Slicerem generován NC program v G-kódu, který je velmi podobný těm, které řídí CNC obráběcí stroje.



Obrázek 55: Grafické zobrazení ve sliceru (38)

Slicery též nabízí grafické rozhraní, ve kterém je možné zobrazit každou vrstvu (obrázek 55) a trajektorii trysky v rovině XY, což je benefitem při ladění výše uvedených parametrů 3D tisku, a tedy k dosažení co nejlepšího výsledku.

G-kód

Nejpoužívanější kódy používané při 3D tisku:

G00, G01 – rychloposuv, lineární interpolace

Příklad použití kódu: G1 X90.6 Y13.8 E22.4

V řádku jsou za číslem kódu souřadnice v X, Y a Z. Na poslední pozici je hodnota pro posuv filamentu v extrudéru v milimetrech. Posuvová rychlost se udává také milimetrech za minutu, hodnota je za písmenem F.

G02, G03 – kruhová interpolace

G04 – Setrvání

G90 – Absolutní pozicování

G91 – Relativní pozicování

G92 – Nastavení pozice – používáno pro nastavení nulové hodnoty pro extrudér, v určitých případech i pro osy

Příklad G-kódu:

G92 E0.0 - nastavení nuly pro extrudér

G90 - Absolutní pozicování

G1 Z0.200 F9000.000 - Posuv osy Z o 0,2 mm rychlostí 9000 mm/min

G1 F1200.000 - Nastavení rychlosti posuvu

G1 X81.521 Y64.024 E0.3552 – Lineární interpolace s určením posunu os X a Y,

G1 X83.782 Y63.387 E0.0736 posunutí filamentu o 0,3552 mm.

G1 X85.845 Y62.988 E0.0658

Součástí G-kódu jsou i informace o množství spotřebovaného filament, době tisku či parametrech nastavených ve Sliceru.

7.4. Popis struktury programu:

Části programu, které se opakují pro všechny osy, jsou popsány jen pro osu X:

Pro napsání programu bylo potřeba používat podmínkování pomocí funkcí IF, ELSE, ELSIF a funkce CASE.

Pomocí funkčních bloků *MC Power* se aktivuje osa a povolí záporný i kladný směr pohybu, *MC Jog* slouží k zapnutí pohybu dopředu či vzad podle použití *JogForward* / *JogBackwards*.

Funkční blok *MC MoveAbsolute* vykoává pohyb osy o vzdálenost, který odpovídá proměnné *DelkaWord*. Všechny tyto funkce lze ovládat z HMI.

IF manual THEN

```
fb_MC_Power.Enable := ax1.Enable;  
fb_MC_Power.Enable_Positive := ax1.Positive;  
fb_MC_Power.Enable_Negative := ax1.Negativ;  
fb_MC_Jog.JogForward := ax1.JogForward;  
fb_MC_Jog.JogBackwards := ax1.JogBackWard;  
fb_MC_MoveAbsolute.Position := DelkaWord;  
fb_MC_MoveAbsolute.Execute := Pozice1;
```

Zároveň je pro rychlejší ovládání na HMI tlačítko, které aktivuje první tři proměnné, z výše uvedené částí kódu.

IF AktivovatOsuX THEN

```
Main.Ax1.Enable := TRUE;  
Main.Ax1.Negativ := TRUE;  
Main.Ax1.Positive := TRUE;  
END_IF;
```

Pokud je aktivována proměnná *Homování*, dojde k rozpohybování osy Z dokud nedojde k sepnutí koncového spínače osy Z, tedy zapsání hodnoty TRUE do proměnné lamp3N. Následně se spustí funkční bloky *MC_Jog.JogBackwards* pro pohyb os X a Y dokud nedojde k sepnutí příslušných koncových snímačů. Pokud je proces dokončen změní se proměnná Homování do hodnoty TRUE.

```
IF Homovani THEN
```

```
    fb_MC_Jog3.JogForward := NOT lamp3n;
```

```
    IF lamp3n THEN
```

```
        fb_MC_Jog.JogBackwards := NOT lamp1n;
```

```
        fb_MC_Jog2.JogBackwards := NOT lamp2n;
```

```
    ELSE
```

```
        fb_MC_Jog.JogBackwards := FALSE;
```

```
        fb_MC_Jog2.JogBackwards := FALSE;
```

```
    END_IF
```

```
    IF lamp1n AND lamp2n AND lamp3n THEN
```

```
        Homovani := FALSE;
```

```
    END_IF
```

```
END_IF
```

Načítání NC kódu se provádí pomocí funkčního bloku *fbSimpleNciSequence*

```
fbSimpleNciSequence(  
    bReset := reset,  
    bExecute:= ZpracujKod,  
    bBusy=> ,  
    bError=> ,  
    bDone=> ,  
    stX:=gvl.ax1,
```

```
stY:=gvl.ax2,  
stZ:=gvl.ax3,  
stItpToPlc:=in_stItpToPlc ,  
stPlcToItp:= out_stPlcToItp);
```

Tento funkční blok používá funkci CASE, kdy dochází k postupnému vykonávání úloh.

```
fbBuildGroup(  
    bExecute:=TRUE,  
    nGroupId:=ItpGetGroupId(sNciToPlc:=stItpToPlc) ,  
    nXAxisId:=stX.NcToPlc.AxisId,  
    nYAxisId:=stY.NcToPlc.AxisId,  
    nZAxisId:=stZ.NcToPlc.AxisId,
```

V prvním kroku jsou aktivovány příslušné proměnné, dále se vytvoří kinematická skupina os X, Y, Z

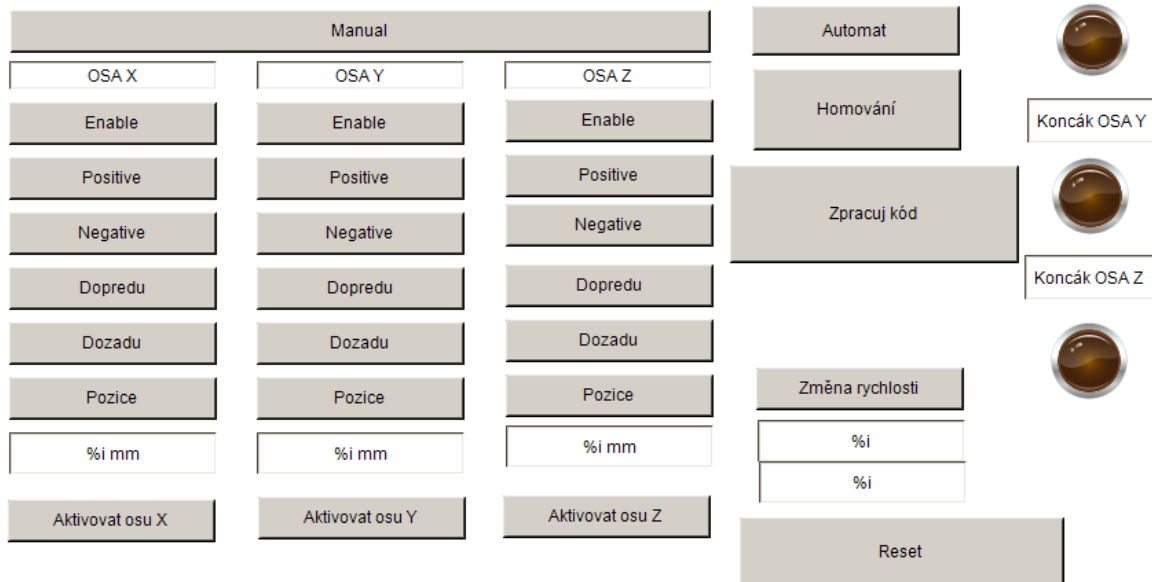
```
fbBuildGroup(  
    bExecute:=TRUE,  
    nGroupId:=ItpGetGroupId(sNciToPlc:=stItpToPlc) ,  
    nXAxisId:=stX.NcToPlc.AxisId,  
    nYAxisId:=stY.NcToPlc.AxisId,  
    nZAxisId:=stZ.NcToPlc.AxisId,
```

Funkční blok *fbLoadGCode* načítá G-kód, uložený ve složce TwinCAT\MC\NCI-folder na disku C: v paměti IPC. Název souboru se přiřazuje do proměnné *sPrgName*. Následně je zkontrolován stav interpretru G-kódu po kterém se aktivuje funkční blok *fbStartCode*. Tímto se aktivuje řetězec NC-program – interpret – drivery – pohony.

7.5. Uživatelské rozhraní

V TwinCat záložce VISUs, je možná tvorba uživatelského rozhraní pro ovládání vytvořeného programu. V zorném poli pracovního okna jsou umístěna tlačítka propojené na proměnné v programu. Pro indikaci najetí do referenční polohy – rozepnutí kontaktu na koncovém spínači, jsou zde indikační diody. Do polí pro určení požadované pozice os

a rychlosti jsou zde pole do kterých lze zadat číselnou hodnotu. Vizualizace je uživatelského rozhraní je na obrázku 56.



Obrázek 56: Uživatelské rozhraní

7.6. Potřebné úkony a hardware pro navazující zprovoznění

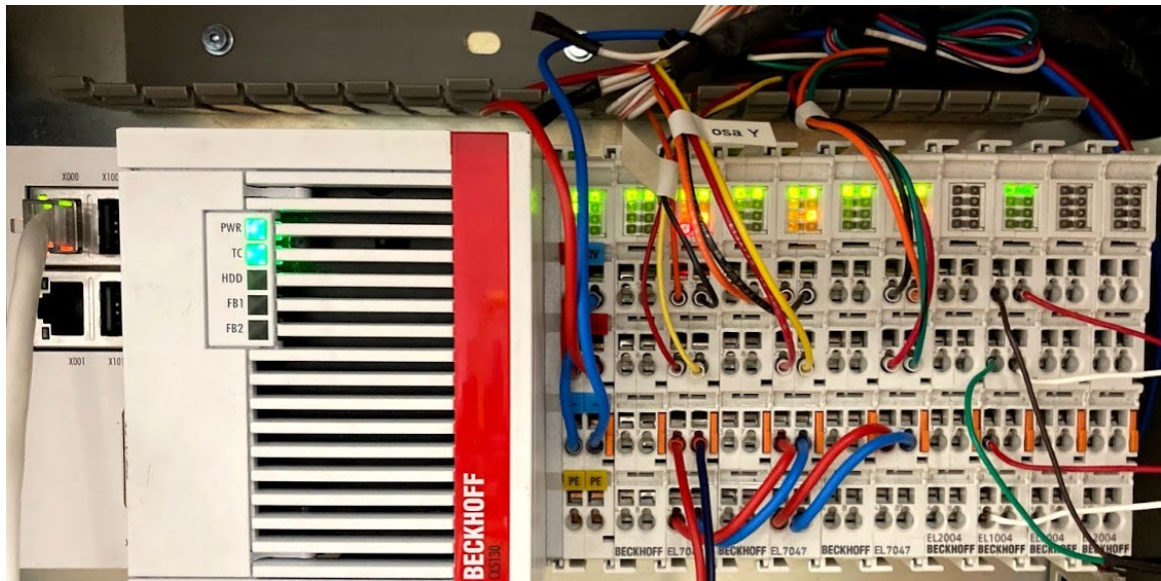
Řízení pohonu extrudéru – tisková hlava tiskárny Stratasys má dva extrudéry pro základní i podpurný materiál. Extrudéry mají pohyb realizovaný 12V DC motorem s enkodérem. Pro řízení lze použít karta Beckhoff EL7342, na straně programu to znamená řízení motoru na základě hodnoty E v G-kódu.

Temperování tiskové komory a trysky extrudéru – v tiskárně jsou pro detekci teploty termočlánky, ty mohou být zapojeny do dvoukanálové vstupní karty EL3312. Spínání topných elementů pak obsluží karta EL2502. Mezi topný element a kartu bude třeba vložit SSR relé s ovládacím napětím 24VDC a spínaným napětím 120VAC. Do programu se přidá PID regulace

Zámky jsou ve dveřích 3D tiskárny a také slouží pro zajištění pozice cartridge filamentu. Pro spínání 24 VDC obvodů mohou být použity 2 karty KL2602. Stejně karty se mohou použít též pro spínání osvětlení a ventilátorů.

Zámky dveří zde zajišťují bezpečnost práce na stroji, nicméně jsou to jen jednobokové solenoidy, které nelze považovat za bezpečnostní prvek. Ten by měl mít v první řadě zdvojené kontakty. Když dojde ke spečení nebo nefunkčnosti jednoho z obvodů, je tu ještě druhý, zároveň jsou bezpečnostní systémy vybavené funkcí EDM - External Device Monitoring, která stav obvodů stále kontroluje. Pokud tedy použijeme bezpečnostní prvky, je třeba použít též bezpečnostní relé, nejen karty digitálních vstupů. Rozdělení bezpečnostních skupin je více, nejpoužívanější je ISO 13849-1 – Performance level. Na výrobních strojích se používají skupiny PLd a PLe, normy mají velké množství parametrů, na základě kterých se bezpečnost vyhodnocuje. Velmi obecně lze však říct, že pokud může zařízení způsobit lehká až střední zranění, použijí se komponenty s bezpečnostní skupinou PLd. Zda může stroj způsobit těžká zranění až zranění neslučitelná se životem, musí být použity komponenty s bezpečnostní skupinou PLe.

Pro realizaci uživatelského rozhraní na fyzickém HMI může být použit libovolný dotykový panel. V případě tlaku na cenu by ovšem bylo možné použít pouze zobrazovač, kdy bude výběr mezi tlačítky ovládan rotačním kolečkem s potvrzovací funkcí. Zapojené IPC s kartami je na obrázku 57.



Obrázek 57: Sestava umístěná v rozvaděči tiskárny Stratasy

8. Závěr

Úvod diplomové práce se zabývá technologiemi aditivní výroby se zaměřením na technologii FDM a 3D tisk obecně. Řízení FDM 3D tiskáren je závislé na typu konstrukčního provedení, které popisuje kapitola 2.1. Dále práce pojednává o možnostech použití řídicího systému, který by mohl nahradit stávající nefunkční systém. Vzhledem k tomu, že hlavní úlohou řídicího systému je realizace interpolovaného pohybu ve třech osách na základě NC programu, zabývá se následující kapitola krokovými motory, jejich konstrukcí a možnostmi řízení.

Kapitola 4 popisuje tiskárnu Stratasys, stavbu původního řídicího systému, komponenty a současný stav 3D tiskárny. Jako náhrada původního řídicího systému byl vzhledem k robustnosti, průmyslovému provedení a vhodným nástrojům zvolen PLC systém Beckhoff. Ve vývojovém prostředí TwinCat byla provedena hardwarová konfigurace. Došlo k nastavení krokových pohonů a napsání programu v podobě strukturovaného textu pro aktivaci os jejich manuálnímu ovládní. Následně byl vytvořen program pro najíždění do referenčních poloh a program pro načítání NC programu v G-kódu. Laserovým měřicím senzorem byly vyhodnoceny převodové poměry os. Grafické zobrazení měřených hodnot pomohlo k naladění krokových motorů. V současném stavu retrofitu 3D tiskárny Stratasys, lze vykonávat interpolovaný pohyb os X, Y a Z, které polohují tiskovou hlavu na základě NC programu.

Navazující prací by mělo být zprovoznění ovládní DC motorů pro řízení extrudéru, temperování tiskové hlavy a tiskové komory.

Největším úskalím tvorby diplomové práce bylo naučit se ovládat funkcionalitu prostředí TwinCat.

9. Seznam obrázků

Obrázek 1: Ovládací panel řídicího systému Sinumeric.....	12
Obrázek 2: Řídicí deska postavená na systému Arduino.....	14
Obrázek 3: Schéma driveru krokového motoru	16
Obrázek 4: The Box od společnosti BLB Industries.....	17
Obrázek 5: Tiskárna RepRap.....	19
Obrázek 6: Štít pro pracovníky ve zdravotnictví.....	19
Obrázek 7: Extrudér FDM tiskárny.....	20
Obrázek 8: Srovnání sestavy s bowdenem (vlevo) a bez bowdenu	21
Obrázek 9: Nákres technologie SLS	23
Obrázek 10: Dílec vyrobený technologií Selective Laser Sintering	24
Obrázek 11: Nákres technologie SLA	24
Obrázek 12: SLA výtisk.....	25
Obrázek 13: Technologie Laminated Object Manufacturing (LOM)	25
Obrázek 14: Kartézské konstrukční uspořádání os.....	26
Obrázek 15: Schéma Delta tiskárny.....	27
Obrázek 16: Geometrie ramene řešení Delta.....	27
Obrázek 17: Tiskárna s kinematikou SCARA.....	28
Obrázek 18: H-bot – Kinematika os X a Y.....	29
Obrázek 19: Tiskárna s kinematikou H-bot	30
Obrázek 20: Krokový motor s permanentním magnetem	31
Obrázek 21: Variabilní reluktance zapojení.....	32
Obrázek 22: Rotor a stator hybridního krokového motoru.....	33
Obrázek 23: Schéma hybridního krokového motoru.....	34
Obrázek 24: Unipolární a bipolární zapojení.....	35
Obrázek 25: Lineární interpolace v kartézském souřadném systému	36
Obrázek 26: Odchylka při lineární interpolaci.....	37
Obrázek 27: Odchylka kruhové interpolace	38
Obrázek 28: Stratasys Dimension Elite/768	40
Obrázek 29: Služby, které řídí procesor Intel 80186.....	41
Obrázek 30: Uspořádání os na tiskárně Stratasys.....	42
Obrázek 31: Příklad pulsní šířkové modulace.....	43
Obrázek 32: Schéma řídicích a napájecích prvků.....	44
Obrázek 33: IPC Beckhoff CX5130	45
Obrázek 34: Hlavní okno programového rozhraní TwinCat	46
Obrázek 35: Sestava IPC s kartami pro řízení os 3D tiskárny.....	48
Obrázek 36: Koncový spínač Hamlin 59085	49
Obrázek 37: Alokace jádra procesoru pro TwinCat	50
Obrázek 38: Skupiny nového projektu.....	50
Obrázek 39: Skupina Motions	51
Obrázek 40: Přiřazení os	52
Obrázek 41: DUT.....	52
Obrázek 42: Modul I/O	54
Obrázek 43: Diagram manuálního režimu.....	55
Obrázek 44: Diagram automatického režimu.....	56
Obrázek 45: Průběhy proudů pro plné kroky	58



Obrázek 46: Průběh proudů při mikrokrokování	59
Obrázek 47: Triangulační laserový senzor OD5000-C85T20	61
Obrázek 48: Průběh měření pohybu os	61
Obrázek 49: Nastavení parametrů 1	62
Obrázek 50: Znatelný překmit při dojetí na pozici.....	62
Obrázek 51: Nastavení parametrů 2.....	63
Obrázek 52: Časový průběh naladěného pohonu	63
Obrázek 53: Prostředí pro ovládání pohonů	64
Obrázek 54: Ovládací okno s nahraným NC programem	64
Obrázek 55: Grafické zobrazení ve sliceru	65
Obrázek 56: Uživatelské rozhraní	70
Obrázek 57: Sestava umístěná v rozvaděči tiskárny Stratasys	72

10. Seznam použité literatury

- 1) SOUČEK, PAVEL. *Pohony výrobních zařízení: Servomechanismy*. Praha: ČVUT, 1997.
- 2) SCHMID, Dietmar. *Řízení a regulace pro strojírenství a mechatroniku*. Praha: Europa - Sobotáles, 2005. ISBN 80-867-0610-9.
- 3) G.Frey a L.Litz. *Formal methods in PLC programming*. Nashville, TN, USA, 06 August 2002n. I. ISBN 0-7803-6583-6
- 4) Nannan GUO a Ming C. LEU. *Additive manufacturing: technology, applications and research needs*. Berlin: Higher Education Press and Springer-Verlag Berlin Heidelberg, 2013.
- 5) Řídicí systém Sinumerik 808D ADVANCED. *První hanácká BOW* [online]. 2018, 23.8.2019 [cit. 2021-8-8]. Dostupné z: <https://www.bow.cz/clanky/ridici-system-sinumerik-808d-advanced-24>
- 6) F-Yi GRBL Controller Board 3 Axis Controller Panel for DIY CNC 3018 1610 2418. *Amazon.ca* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.amazon.ca/F-Yi-GRBL-Controller-Board-Panel/dp/B07TWB6W9S>
- 7) Phill Leyva. BIPOLAR STEPPER-MOTOR DRIVER HAS FEW COMPONENTS. *Maxim Integrated* [online]. 2009 [cit. 2021-8-8]. Dostupné z: <https://www.maximintegrated.com/en/design/technical-documents/app-notes/4/4469.html>
- 8) The Best Large FDM 3D Printers. *3D prints.com* [online]. 2019 [cit. 2021-8-8]. Dostupné z: <https://3d-prints.com/2019/01/10/the-best-large-fdm-3d-printers-of-2019/>
- 9) RepRapPro Mendel/cs. *RepRap* [online]. 2020 [cit. 2021-8-8]. Dostupné z: https://reprap.org/wiki/RepRapPro_Mendel/cs
- 10) Kompletace Prusa obličejového štítu - RC1/RC2/RC3. *Prusa Knowledge Base* [online]. [cit. 2021-8-8]. Dostupné z: https://help.prusa3d.com/cs/guide/kompletace-prusa-oblicejoveho-stitu-rc1-rc2-rc3_125495
- 11) *Beckhoff Information System* [online]. Beckhoff Automation, 2021 [cit. 2021-8-8]. Dostupné z: <https://infosys.beckhoff.com/>
- 12) 3D Printing Concepts and 3D Printer Parts. *LeapFrog* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.lpfrg.com/guides/3d-printing-concepts-and-3d-printer-parts/>

- 13) *Direct Drive, Bowden, Remote Motor, the differences* [online]. In: . Qualup SAS [cit. 2021-8-8]. Dostupné z: <http://www.spiderbot.eu/direct-drive-bowden-remote-motor-the-differences/?lang=en>
- 14) FDM. *DRUCKWEGE 3D PRINTING* [online]. [cit. 2021-8-8]. Dostupné z: <https://druckwege.de/en/home-en/technology/fused-deposition-modelling-fdm>
- 15) 3D PRINTER BED ADHESION. *All3DP* [online]. [cit. 2021-8-8]. Dostupné z: <https://all3dp.com/2/3d-printer-bed-adhesion-all-you-need-to-know/>
- 16) Technologie 3D tisku. *EasyCNC* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.easycnc.cz/inpage/informace-o-technologiech-3d-tisku/>
- 17) The Complete Guide to Stereolithography (SLA) in 3D Printing. *3Dnatives* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.3dnatives.com/en/stereolithography-explained100420174/>
- 18) The Best Industrial 3D Printers. *Business.com* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.business.com/categories/best-industrial-3d-printers/>
- 19) Efficient use of 3D printing in metal ready to hand. *Sirris* [online]. 18.8.2017 [cit. 2021-8-8]. Dostupné z: <https://sirris.be/nl/node/48790>
- 20) Dimension Elite/768. *Computer Aided Technology* [online]. [cit. 2021-8-8]. Dostupné z: <https://store.cati.com/Catalog/Printers/Detail/403-180-00106-dimension-elite768>
- 21) PLCs Explained: Programmable Logic Controller Basics. In: *Trimantec* [online]. TecTalk [cit. 2021-8-8]. Dostupné z: <https://trimantec.com/blogs/t/plcs-explained>
- 22) SUL, Seung-Ki. *Control of electric machine drive system*. Wiley-IEEE, 2011. ISBN 97 804 7059 0799.
- 23) *Dimension® 768/Elite: Service Manual*. USA: Stratasys, Incorporated, 2010.
- 24) *TF5100: Manual TwinCAT 3 | NC I*. Beckhoff Automation, 2021.
- 25) VACULÍK, Martin. *Rapid prototyping*. Zlín, 2011. Bakalářská práce. Univerzita Tomáši Bati ve Zlíně. Vedoucí práce Ing. Michal Staněk, Ph.D.
- 26) BETINA MADEIRA SCHMITT, CHRISTIANO FRAGA ZIRBES a CASSIANO BONIN. A Comparative Study of Cartesian and Delta 3D Printers on Producing PLA Parts. *Scielo Brazil* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.scielo.br/j/mr/a/h5FWSSJ7f9WZbwtSVpG3jBQ/>



- 27) BELL, Charles. *Delta Printer Hardware*. USA: Apress, Berkeley, CA, 2005. ISBN 978-1-4842-1173-1.
- 28) 3D POTTERBOT SCARA V4. *Bimetry.shop* [online]. [cit. 2021-8-8]. Dostupné z: <https://bimetry.shop/downloads/3d-potterbot-scara-v4/>
- 29) CoreXY vs. H-Bot: Which System is Better for a 3D Printer? *3D Printer Bee* [online]. [cit. 2021-8-8]. Dostupné z: <https://the3dprinterbee.com/corexy-vs-hbot/>
- 30) Krokové motory. *Profirobot* [online]. 2013 [cit. 2021-8-8]. Dostupné z: <http://profirobot.cz/wp-content/uploads/2018/03/Krokov%C3%A9-motory.pdf>
- 31) KIRAN DAWARE. Basics Of Stepper Motor. *Electriceasy* [online]. 2014 [cit. 2021-8-8]. Dostupné z: <https://www.electriceasy.com/2014/10/stepper-motor.html>
- 32) Unipolar/Bipolar Connections (2-Phase Motors). *Motion King* [online]. MotionKing Motor Industry Co., 2015 [cit. 2021-8-8]. Dostupné z: http://www.motionking.com/support/unipolar_bipolar.htm
- 33) CX5130 | Embedded PC with Intel Atom® processor. *Beckhoff Automation* [online]. 2021 [cit. 2021-8-8]. Dostupné z: <https://www.beckhoff.com/en-en/products/ipc/embedded-pcs/cx5100-intel-atom/cx5130.html>
- 34) EL7047 | Stepper motor terminal 48 V DC, 5 A, with incremental encoder, vector control. *Beckhoff Automation* [online]. 2021 [cit. 2021-8-8]. Dostupné z: <https://www.beckhoff.com/en-en/products/i-o/ethercat-terminals/el7xxx-compact-drive-technology/el7047.html>
- 35) Schindler Elevator QKS9 door machine speed sensor photoelectric switch HAMLIN 59085-502 AQ1H148. *Imall* [online]. [cit. 2021-8-8]. Dostupné z: <https://imall.com/product/1pcs-Schindler-Elevator-QKS9-door-machine-speed-sensor-photoelectric-switch-HAMLIN-59085-502-AQ1H148/Tools-Tool-Parts/aliexpress.com/4001001977921/144-50318001/en> [online]. [cit. 2021-8-8].
- 36) GHEORGHE BALUTA. Microstepping Mode for Stepper Motor Control. *IEEE Explore* [online]. 2007 [cit. 2021-8-8]. Dostupné z: <https://ieeexplore.ieee.org/document/4292799>
- 37) Displacement measurement sensors OD5000. *SICK* [online]. [cit. 2021-8-8]. Dostupné z: <https://www.sick.com/ag/en/distance-sensors/displacement-measurement-sensors/od5000/od5000-c85t20/p/p516744>
- 38) 3D Printer Software: What You Need To Know. Dremel DigiLab [online]. [cit. 2021-8-8]. Dostupné z: <https://digilab.dremel.com/blog/3d-printer-software-what-you-need-know>

11. Přílohy

Kódy a proměnné:

DUTs:

```
TYPE DUT_I_MyAxis :  
STRUCT  
    Enable: BOOL;  
    Positive: BOOL;  
    Negativ: BOOL;  
    JogForward: BOOL;  
    JogBackWard: BOOL;  
END_STRUCT  
END_TYPE
```

GVLs:

```
{attribute 'qualified_only'}  
VAR_GLOBAL  
    ax1: AXIS_REF;  
    ax2: AXIS_REF;  
    ax3: AXIS_REF;  
END_VAR
```



Proměnné MAIN:

PROGRAM MAIN

VAR

```
Ax1: DUT_I_MyAxis;  
Ax2: DUT_I_MyAxis;  
Ax3: DUT_I_MyAxis;  
Manual : BOOL;  
Automat : BOOL;  
Lamp1 AT %I*: BOOL;  
Lamp2 AT %I*: BOOL;  
Lamp3 AT %I*: BOOL;  
Lamp1N : BOOL;  
Lamp2N : BOOL;  
Lamp3N : BOOL;  
Homovani : BOOL;  
Pozice1 : BOOL;  
Pozice2 : BOOL;  
Pozice3 : BOOL;  
DelkaText : STRING;  
DelkaWord: INT;  
DelkaWord2: INT;  
DelkaWord3: INT;  
Reset: BOOL;  
RychlostNum : INT;  
RychlostAktualniNum: INT;  
RychlostZmenit: BOOL;  
Velocity1: INT;  
Velocity2: INT;  
Velocity3: INT;  
AktivovatOsuX: BOOL;
```




AktivovatOsuY: BOOL;

AktivovatOsuZ: BOOL;

fb_MC_Power: MC_Power;

fb_MC_Jog: MC_Jog;

fb_MC_Power2: MC_Power;

fb_MC_Jog2: MC_Jog;

fb_MC_Power3: MC_Power;

fb_MC_Jog3: MC_Jog;

fb_MC_MoveAbsolute: MC_MoveAbsolute;

fb_MC_MoveAbsolute2: MC_MoveAbsolute;

fb_MC_MoveAbsolute3: MC_MoveAbsolute;

fb_MC_Reset1: MC_Reset;

fb_MC_Reset2: MC_Reset;

fb_MC_Reset3: MC_Reset;

ZpracujKod: BOOL;

fbSimpleNciSequence: FB_SimpleNciSequence;

bExecSimpleNci: BOOL := FALSE;

in_stItpToPlcAT %I*: NcToPlc_NciChannel_Ref;

out_stPlcToItp AT %Q*: PLCTONC_NCICHANNEL_REF;

bUserEnableAxes: BOOL := TRUE;

fUserOverride: LREAL := 100.0;

END_VAR

Program MAIN:

IF AktivovatOsuX THEN

 Main.Ax1.Enable := TRUE;

 Main.Ax1.Negativ := TRUE;

 Main.Ax1.Positive := TRUE;

END_IF;

IF AktivovatOsuY THEN

 Main.Ax2.Enable := TRUE;

 Main.Ax2.Negativ := TRUE;

 Main.Ax2.Positive := TRUE;

END_IF;

IF AktivovatOsuZ THEN

 Main.Ax3.Enable := TRUE;

 Main.Ax3.Negativ := TRUE;

 Main.Ax3.Positive := TRUE;

END_IF;

IF RychlostZmenit THEN

 fbSimpleNciSequence.iAxisOvr := INT_TO_UDINT(RychlostNum)*10000;

 RychlostZmenit := FALSE;

END_IF;

RychlostAktualniNum

:=

UDINT_TO_INT(fbSimpleNciSequence.oAxisOvr/10000);

Lamp1N := NOT Lamp1;

Lamp2N := NOT Lamp2;

Lamp3N := NOT Lamp3;



IF manual THEN

```
fb_MC_Power.Enable := ax1.Enable;  
fb_MC_Power.Enable_Positive := ax1.Positive;  
fb_MC_Power.Enable_Negative := ax1.Negativ;  
fb_MC_Jog.JogForward := ax1.JogForward;  
fb_MC_Jog.JogBackwards := ax1.JogBackWard;  
fb_MC_MoveAbsolute.Position := DelkaWord;  
fb_MC_MoveAbsolute.Execute := Pozice1;  
  
fb_MC_Power2.Enable := ax2.Enable;  
fb_MC_Power2.Enable_Positive := ax2.Positive;  
fb_MC_Power2.Enable_Negative := ax2.Negativ;  
fb_MC_Jog2.JogForward := ax2.JogForward;  
fb_MC_Jog2.JogBackwards := ax2.JogBackWard;  
  
fb_MC_Power3.Enable := ax3.Enable;  
fb_MC_Power3.Enable_Positive := ax3.Positive;  
fb_MC_Power3.Enable_Negative := ax3.Negativ;  
fb_MC_Jog3.JogForward := ax3.JogForward;  
fb_MC_Jog3.JogBackwards := ax3.JogBackWard;
```

ELSIF automat THEN

```
fb_MC_Power.Enable := TRUE;  
fb_MC_Power2.Enable := TRUE;  
fb_MC_Power3.Enable := TRUE;  
  
fb_MC_Power.Enable_Positive := TRUE;  
fb_MC_Power2.Enable_Positive := TRUE;  
fb_MC_Power3.Enable_Positive := TRUE;  
  
fb_MC_Power.Enable_Negative := TRUE;
```



```
fb_MC_Power2.Enable_Negative := TRUE;
fb_MC_Power3.Enable_Negative := TRUE;

IF Homovani THEN
    fb_MC_Jog3.JogForward := NOT lamp3n;
    IF lamp3n THEN
        fb_MC_Jog.JogBackwards := NOT lamp1n;
        fb_MC_Jog2.JogBackwards := NOT lamp2n;
    ELSE
        fb_MC_Jog.JogBackwards := FALSE;
        fb_MC_Jog2.JogBackwards := FALSE;
    END_IF
END_IF

    IF lamp1n AND lamp2n AND lamp3n THEN
        Homovani := FALSE;
    END_IF
END_IF
ELSE
    fb_MC_Power.Enable := FALSE;
    fb_MC_Power.Enable_Positive := FALSE;
    fb_MC_Power.Enable_Negative := FALSE;
    fb_MC_Jog.JogForward := FALSE;
    fb_MC_Jog.JogBackwards := FALSE;
    ax1.Enable := FALSE;
    ax1.Positive := FALSE;
    ax1.Negativ := FALSE;
    ax1.JogForward := FALSE;
    ax1.JogBackWard := FALSE;

    fb_MC_Power2.Enable := FALSE;
    fb_MC_Power2.Enable_Positive := FALSE;
```



```
fb_MC_Power2.Enable_Negative := FALSE;  
fb_MC_Jog2.JogForward := FALSE;  
fb_MC_Jog2.JogBackwards := FALSE;  
ax2.Enable := FALSE;  
ax2.Positive := FALSE;  
ax2.Negativ := FALSE;  
ax2.JogForward := FALSE;  
ax2.JogBackWard := FALSE;
```

```
fb_MC_Power3.Enable := FALSE;  
fb_MC_Power3.Enable_Positive := FALSE;  
fb_MC_Power3.Enable_Negative := FALSE;  
fb_MC_Jog3.JogForward := FALSE;  
fb_MC_Jog3.JogBackwards := FALSE;  
ax3.Enable := FALSE;  
ax3.Positive := FALSE;  
ax3.Negativ := FALSE;  
ax3.JogForward := FALSE;  
ax3.JogBackWard := FALSE;
```

END_IF

IF reset THEN

```
fb_MC_Reset1.Execute := TRUE;  
fb_MC_Reset2.Execute := TRUE;  
fb_MC_Reset3.Execute := TRUE;  
reset := false;
```

ELSE

```
fb_MC_Reset1.Execute := FALSE;  
fb_MC_Reset2.Execute := FALSE;  
fb_MC_Reset3.Execute := FALSE;
```

END_IF



fb_MC_Power(

Axis:= gvl.ax1,
Override:= ,
BufferMode:= ,
Options:= ,
Status=> ,
Busy=> ,
Active=> ,
Error=> ,
ErrorID=>);

fb_MC_Jog(

Axis:= gvl.ax1,
Mode:= ,
Position:= ,
Velocity:=100 ,
Acceleration:= ,
Deceleration:= ,
Jerk:= ,
Done=> ,
Busy=> ,
Active=> ,
CommandAborted=> ,
Error=> ,
ErrorID=>);

fb_MC_Power2(

Axis:= gvl.ax2,
Override:= ,
BufferMode:= ,
Options:= ,



```
Status=> ,  
Busy=> ,  
Active=> ,  
Error=> ,  
ErrorID=> );
```

```
fb_MC_Jog2(  
  Axis:= gvl.ax2,  
  Mode:= ,  
  Position:= ,  
  Velocity:= fbSimpleNciSequence.oAxisOvr,  
  Acceleration:= ,  
  Deceleration:= ,  
  Jerk:= ,  
  Done=> ,  
  Busy=> ,  
  Active=> ,  
  CommandAborted=> ,  
  Error=> ,  
  ErrorID=> );
```

```
fb_MC_Power3(  
  Axis:= gvl.ax3,  
  Override:= ,  
  BufferMode:= ,  
  Options:= ,  
  Status=> ,  
  Busy=> ,  
  Active=> ,
```



```
Error=> ,  
ErrorID=> );
```

```
fb_MC_Jog3(  
  Axis:= gvl.ax3,  
  Mode:= ,  
  Position:= ,  
  Velocity:= fbSimpleNciSequence.oAxisOvr ,  
  Acceleration:= ,  
  Deceleration:= ,  
  Jerk:= ,  
  Done=> ,  
  Busy=> ,  
  Active=> ,  
  CommandAborted=> ,  
  Error=> ,  
  ErrorID=> );
```

```
fb_MC_MoveAbsolute(  
  Axis:= gvl.ax1 ,  
  Velocity:= fbSimpleNciSequence.oAxisOvr,  
  Acceleration:= ,  
  Deceleration:= ,  
  Jerk:= ,  
  BufferMode:= ,  
  Options:= ,  
  Done=> ,  
  Busy=> ,  
  Active=> ,  
  CommandAborted=> ,  
  Error=> ,
```




ErrorID=>);

fb_MC_MoveAbsolute2(

Axis:= gvl.ax2,

Velocity:= fbSimpleNciSequence.oAxisOvr,

Acceleration:= ,

Deceleration:= ,

Jerk:= ,

BufferMode:= ,

Options:= ,

Done=> ,

Busy=> ,

Active=> ,

CommandAborted=> ,

Error=> ,

ErrorID=>);

fb_MC_MoveAbsolute3(

Axis:= gvl.ax3 ,

Velocity:= fbSimpleNciSequence.oAxisOvr,

Acceleration:= ,

Deceleration:= ,

Jerk:= ,

BufferMode:= ,

Options:= ,

Done=> ,

Busy=> ,

Active=> ,

CommandAborted=> ,

Error=> ,



```
ErrorID=> );
```

```
//ltpSetOverridePercent(fOverridePercent:=fUserOverride  
sPlcToNci:=out_stPlcToltp);
```

```
fbSimpleNciSequence(  
bReset := reset,  
bExecute:= ZpracujKod,  
bBusy=> ,  
bError=> ,  
bDone=> ,  
stX:=gvl.ax1,  
stY:=gvl.ax2,  
stZ:=gvl.ax3,  
stltpToPlc:=in_stltpToPlc ,  
stPlcToltp:= out_stPlcToltp);
```

```
fb_MC_Reset1(  
Axis:=gvl.ax1 ,  
Done=> ,  
Busy=> ,  
Error=> ,  
ErrorID=> );
```

```
fb_MC_Reset2(  
Axis:=gvl.ax2 ,  
Done=> ,  
Busy=> ,  
Error=> ,  
ErrorID=> );
```



```
fb_MC_Reset3(  
Axis:=gvl.ax3 ,  
Done=> ,  
Busy=> ,  
Error=> ,  
ErrorID=> );
```



Proměnné FB_SimpleNCISequence:

FUNCTION_BLOCK FB_SimpleNciSequence

VAR_INPUT

bExecute : BOOL;
bReset : BOOL;
iAxisOvr : UDINT;

END_VAR

VAR_OUTPUT

bBusy : BOOL;
bError : BOOL;
bDone : BOOL;
oAxisOvr : UDINT;

END_VAR

VAR_IN_OUT

stX : AXIS_REF;
stY : AXIS_REF;
stZ : AXIS_REF;
stItpToPlc : NCTOPLC_NCICHANNEL_REF;
stPlcToItp : PLCTONC_NCICHANNEL_REF;

END_VAR

VAR

nState : UDINT := 0;
nErrorState : UDINT;

fbBuildGroup : CfgBuild3DGroup;
fbLoadGCode : ItpLoadProgEx;
fbStartGCode : ItpStartStopEx;
fbClearGroup : CfgReconfigGroup;
fbConfirmHsk : ItpConfirmHsk;



```
nInterpreterState : UDINT := 0;
bConfirmHsk       : BOOL  := FALSE;

sPrgName          : STRING(255) := 'first.nc';
END_VAR
```

Program FB_SimpleNciSequance:

```
stPlcToItp.ChnAxesOvr := iAxisOvr;
oAxisOvr := stPlcToItp.ChnAxesOvr;
CASE nState OF
0:
    IF bExecute THEN
        bBusy := TRUE;
        bDone:= FALSE;
        bError:= FALSE;
        nErrorState := nState;
        nState:= 10;
    END_IF
    IF NOT bReset THEN
        fbClearGroup(bExecute:=FALSE);
    END_IF
10:
    // vytvoření kinematické skupiny
    fbBuildGroup(
        bExecute:=TRUE,
        nGroupId:=ItpGetGroupId(sNciToPlc:=stItpToPlc) ,
        nXAxisId:=stX.NcToPlc.AxisId,
```



```
nYAxisId:=stY.NcToPlc.AxisId,
nZAxisId:=stZ.NcToPlc.AxisId,
tTimeout:= ,
bBusy=> ,
bErr=> ,
nErrId=> );
IF NOT fbBuildGroup.bBusy THEN
  IF NOT fbBuildGroup.bErr THEN
    nState:= 20;
  ELSE
    (* add error handling *)
    nErrorState := nState;
    bBusy := FALSE;
    bError := TRUE;
    nState:= 9999;
  END_IF
  fbBuildGroup(bExecute:=FALSE);
END_IF

20:
//nahraje soubor s G-kódem
// je třeba se aby byl požadovaný NC program ve složce: C:
\TwinCAT\MC\NCI-folder
fbLoadGCode(
  sNciToPlc:=stItpToPlc,
  bExecute:=TRUE,
  sPrg:= sPrgName,
  nLength:= INT_TO_UDINT(LEN(sPrgName)),
  tTimeout:= ,
  bBusy=> ,
  bErr=> ,
```



```
        nErrId=> );  
IF NOT fbLoadGCode.bBusy THEN  
    IF NOT fbLoadGCode.bErr THEN  
        nState:= 30;  
    ELSE  
        (* add error handling *)  
        nErrorState := nState;  
        bBusy := FALSE;  
        bError := TRUE;  
        nState:= 9999;  
    END_IF  
    fbLoadGCode(sNciToPlc:=stltpToPlc, bExecute:=FALSE);  
END_IF
```

30:

// je třeba zkontrolovat stav interpretu, který umožňuje správné vykonávání příkazů G-kódu

```
nInterpreterState := ltpGetStateInterpreter(sNciToPlc:=stltpToPlc);  
IF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_READY THEN  
    nState:= 40;  
ELSE  
    (* add error handling *)  
    nErrorState := nState;  
    bBusy := FALSE;  
    bError := TRUE;  
    nState:= 9999;  
END_IF
```

40:

```
// Start načítání G-kódu  
fbStartGCode(  

```



```
        bStart:=TRUE,
        bStop:=FALSE,
        tTimeout:= ,
        sNciToPlc:= stltpToPlc,
        bBusy=> ,
        bErr=> ,
        nErrId=> );
IF NOT fbStartGCode.bBusy THEN
    IF NOT fbStartGCode.bErr THEN
        nState:= 50;
    ELSE
        (* add error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        nState:= 9999;
    END_IF
    fbStartGCode( bStart:=FALSE, sNciToPlc:= stltpToPlc );
END_IF

50:
nInterpreterState := ltpGetStateInterpreter(sNciToPlc:=stltpToPlc);
IF nInterpreterState <> Tc2_NCI.NCI_INTERPRETER_READY THEN
    // nci is running
    nState:= 60;
END_IF

60:
nInterpreterState := ltpGetStateInterpreter(sNciToPlc:=stltpToPlc);
IF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_READY THEN
    // program has finished
    nState:= 70;
```




```
        ELSIF nInterpreterState = Tc2_NCI.NCI_INTERPRETER_ABORTED
THEN
        // a run-time error occurred - this could be a lag error or something
else...
        // add error handling
        ;
END_IF
70:
// Program skončil
// Uknočit spojení kinematické skupiny
fbClearGroup(
    bExecute:=TRUE,
    nGroupId:=ItpGetGroupId(sNciToPlc:=stItpToPlc) ,
    tTimeOut:= ,
    bBusy=> ,
    bErr=> ,
    nErrId=> );
IF NOT fbClearGroup.bBusy THEN
    IF NOT fbClearGroup.bErr THEN
        nState:= 80;
    ELSE
        (* add error handling *)
        nErrorState := nState;
        bBusy := FALSE;
        bError := TRUE;
        nState:= 9999;
    END_IF
    fbClearGroup(bExecute:=FALSE);
END_IF
80:
bDone:= TRUE;
```



```
        bBusy := FALSE;
        IF NOT bExecute THEN
            nState:= 0;
        END_IF
9999:
    // error state
    IF NOT bExecute THEN
        nState:= 0;
        bError := FALSE;
    END_IF
END_CASE

fbConfirmHsk(
    bExecute:=bConfirmHsk ,
    sNciToPlc:=stltpToPlc ,
    sPlcToNci:=stPlcToItp ,
    bBusy=> ,
    bErr=> ,
    nErrId=> );

IF bReset THEN
    fbClearGroup(
        bExecute:=TRUE,
        nGroupId:=ItpGetGroupId(sNciToPlc:=stltpToPlc) ,
        tTimeOut:= ,
        bBusy=> ,
        bErr=> ,
        nErrId=> );

    nState:= 0;
END_IF
```

Stavové diagramy:

