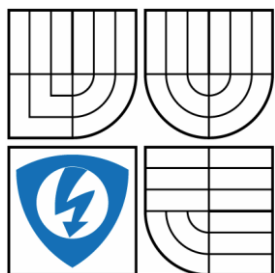


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## MODEL CNC FRÉZKY

MODEL OF CNC MILLING MACHINE

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

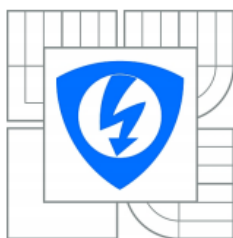
Bc. PAVEL KOVÁŘ

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. RADEK ŠTOHL, Ph.D.

BRNO 2012



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Diplomová práce

magisterský navazující studijní obor  
Kybernetika, automatizace a měření

**Student:** Bc. Pavel Kovář  
**Ročník:** 2

**ID:** 106532  
**Akademický rok:** 2011/2012

**NÁZEV TÉMATU:**

**Model CNC frézky**

## POKYNY PRO VYPRACOVÁNÍ:

1. Nastudujte řídicí příkazy pro tvorbu DPS ve vývojovém systému EAGLE.
2. Implementujte příkazy G-kódu pro CNC frézku do řídicího systému modelu manipulátoru.
3. Vytvořte aplikaci mezi PC a PLC.
4. Ověřte funkčnost systému.

## DOPORUČENÁ LITERATURA:

Kinetix 2000 Multi-axis Servo Drive. (User Manual) Milwaukee: Rockwell Automation, Inc. 2007. 226 s.  
Logix5000 Controllers General Instructions (Reference Manual). Milwaukee: Rockwell Automation, Inc. 2008.

Dle vlastního literárního průzkumu a doporučení vedoucího práce.

**Termín zadání:** 6.2.2012

**Termín odevzdání:** 21.5.2012

**Vedoucí práce:** Ing. Radek Štohl, Ph.D.

**Konzultanti diplomové práce:**

**doc. Ing. Václav Jirsík, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

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

## **Abstrakt**

Diplomová práce se zabývá základními částmi a principy CNC strojů se zaměřením na CNC frézky.

Dále se projekt zabývá popisem manipulátoru, který se nachází v laboratoři E132 v objektu na adrese Kolejní 4, Brno. Práce obsahuje také popis jednotlivých zařízení, která se v modelu manipulátoru nachází.

Nalezneme zde také popis programu generujícího G-kód z obrazu, který si vytvoříme v jeho editoru.

Další část popisuje možné úpravy manipulátoru pro jeho přestavbu na CNC frézku.

Poslední kapitola se zabývá popisem vytvořeného programu na ovládání manipulátoru jako CNC frézky.

## **Klíčová slova**

CNC frézka, Model manipulátoru, PLC, G-kód

## **Abstract**

The master's thesis deals with the basic parts and principles of CNC machines with a focus on a CNC milling machine. There are also comparisons of several CNC machines sold.

Furthermore, the project deals with the description of the manipulator, which is situated in laboratory E-132 in area Kolejní 4, Brno. The work also includes a description of the device, which is located in the manipulator model.

We can find there a description of the program generating G-code from the image that you create in the editor.

The next section describes possible modifications of the manipulator for its reconstruction on a CNC milling machine.

The last chapter deals with the description of program developed for controlling the manipulator as a CNC milling machine.

## **Keywords**

CNC milling machine, Manipulator model, PLC, G-kode

### **Bibliografická citace:**

KOVÁŘ, P. *Model CNC frézky*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 76s. Vedoucí diplomové práce Ing. Radek Štohl, Ph.D.

## **Prohlášení**

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

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

V Brně dne: **21. května 2012**

.....  
podpis autora

## **Poděkování**

Děkuji vedoucímu diplomové práce Ing. Radku Štohlovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne: **21. května 2012**

.....  
podpis autora

# Obsah

1	Úvod .....	10
2	CNC stroje .....	11
2.1	Rozdělení CNC .....	11
2.1.1	CNC soustruhy .....	12
2.1.2	CNC frézky .....	12
2.1.3	CNC obráběcí centra .....	12
2.2	Programování CNC strojů [2],[3] .....	12
2.2.1	G-kód .....	13
2.2.2	M-kód .....	13
2.3	CNC Frézky .....	14
2.3.1	Konstrukce [5] .....	14
2.3.2	Souřadný systém .....	15
2.3.3	Přesnost .....	16
2.3.4	Nástroje .....	16
2.3.5	Ostatní parametry .....	17
3	Popis manipulátoru [12], [13] .....	18
3.1	PLC ControlLogix [15] .....	19
3.2	Procesory 1756-L62S a 1756-LSP [16] .....	20
3.3	Ethernet/IP modul [18] .....	21
3.4	DeviceNet modul [20],[21] .....	22
3.5	SERCOS interface [14],[23] .....	23
3.6	Kinetix 2000 [24],[25] .....	24
3.7	Servomotory [26],[27] .....	25
4	Úpravy Manipulátoru .....	26
4.1	Ochranné kryty .....	26
4.2	Stůl na upnutí .....	26
4.3	Výměna nástroje .....	27
4.4	Rotace nástroje .....	27
4.5	Upevnění frézky .....	28
4.6	Automatické spouštění .....	29
4.7	Kalibrace nástroje .....	30
4.8	Určování nulových bodů .....	30
5	Generování G-kódu .....	31

5.1	Výběr vhodného softwaru [31] .....	31
5.2	Popis programu PCB-GCODE [32] .....	31
5.2.1	Spuštění Eaglu a vytvoření obrazu [33] .....	32
5.2.2	Inicializace PCB GCODE .....	34
5.3	Ostatní programy .....	38
5.3.1	Image to G-kode [41] .....	38
5.3.2	DeskEgrave [42] .....	39
5.3.3	B2G [43] .....	39
6	Přenos dat do PLC .....	40
6.1	DDE server [44], [45] .....	40
6.2	Nastavení DDE serveru [46] .....	40
6.3	Program v MS Excel [46] .....	43
6.3.1	Výběr souboru s kódem .....	43
6.3.2	Načtení vybraného souboru .....	44
6.3.3	Zapsání instrukcí do PLC .....	44
6.3.4	Smazání načtených instrukcí .....	44
7	Program v PLC .....	45
7.1	Konfigurace PLC a komunikace .....	45
7.2	Instrukce pro servoměniče [30] .....	45
7.2.1	MARS .....	46
7.2.2	MSO .....	46
7.2.3	MAH .....	46
7.2.4	MCLM .....	47
7.2.5	MCCM .....	48
7.2.6	MAJ .....	50
7.2.7	MAS .....	51
7.2.8	MSF .....	52
7.3	Popis programu .....	53
7.3.1	Hlavní smyčka .....	53
7.3.2	Ruční režim .....	54
7.3.3	Dekódování instrukce .....	54
7.3.4	Kontrola instrukcí .....	56
7.3.5	Automatický režim .....	57
7.3.6	Kontrola chyb .....	58
7.3.7	Program v safety procesoru .....	59



7.4	Vizualizace [47] .....	59
7.4.1	Úvodní stránka .....	60
7.4.2	Ruční režim .....	60
7.4.3	Programovací režim .....	61
7.4.4	Automatický režim.....	62
7.4.5	Alarmy .....	62
8	Ukázkový program .....	64
8.1	Prostředí Eagle .....	64
8.2	Nastavení PCB-GCODE .....	65
8.3	Nastavení manipulátoru .....	66
9	Závěr.....	68

# 1 ÚVOD

Tato diplomová práce se zabývá CNC stroji, hlavně CNC frézkami. Jejím hlavním úkolem je přestavba modelu manipulátoru, který se nachází v laboratoři E132 v objektu na adrese Kolejní 4, Brno a vytvoření řídicího programu tak, aby se manipulátor choval jako CNC frézka. Kromě programu pro řízení manipulátoru se bude také zabývat řešením generování instrukcí pro obrábění a vytvořením operátorského rozhraní.

V první části projektu bych se chtěl zabývat seznámením s CNC stroji, jejich rozdělením a popisem základních částí se zaměřením na CNC frézky. Začátek kapitoly věnuji rozdělení těchto strojů a dále se chci zmínit o způsobu programování. Rád bych také popsal nejdůležitějších částí, jako jsou základní konstrukce, souřadný systém, nástroje a jejich upínání.

Další kapitola bude věnována popisu modelu manipulátoru a hlavně jeho řídicím komponentům. Po krátkém zevrubném popisu bude následovat část věnovaná řídicímu systému, který se skládá z PLC ControlLogix osazeným dvojicí procesorů, první pro vykonávání samotného programu řízení a druhý pro program zajišťující bezpečnost aplikace. Dále budou popsány komunikační karty pro rozhraní Ethernet/IP, DeviceNet a Sercos. Nebude zde chybět ani popis použitých servoměničů a samotných motorů pro pohon jednotlivých os.

V následující kapitole nalezneme popis a postup změn nezbytných k tomu, aby se model manipulátoru mohl chovat jako CNC frézka. Navrhované změny se týkají jak zajištění bezpečnosti ochrannými kryty, tak i zajištění základní funkčnosti pro obrábění. Mezi navrhované změny patří návrhy na upínací zařízení pro obrobek a nástroj. Dále řešení kalibrace, rotace a výměny nástroje.

V jedné z posledních kapitol budeme rozebírat podrobný postup generování G-kódu pro naši CNC frézku. Popíšeme si zde prostředí, ve kterém se bude vytvářet návrh našeho vzoru pro gravírování a také program, který bude kód generovat. Následovat bude popis transportu vygenerovaného souboru s instrukcemi do paměti PLC.

V předposlední kapitole uvedu popis vytvořeného programu umožňující použití manipulátoru jako CNC frézky. Program bude disponovat režimy pro jeho ovládání jak v ručním, tak i v automatickém režimu. Nedílnou součástí programového vybavení bude také vytvoření rozhraní operátora.

V poslední kapitole si krok za krokem popíšeme postup pro použití vytvořeného prostředí. Na konkrétním příkladu si ukážeme nastavení nejdůležitějších náležitostí nezbytných pro bezproblémový chod naší CNC frézky. Postupně se dozvíme, jak se jednotlivé nástroje používají a k čemu slouží. Tento postup bude začínat instalací programového vybavení a končit vytvořením požadovaného vzoru.

## 2 CNC STROJE

Pokud mluvíme o CNC strojích [38],[39], v naprosté většině si představíme CNC obráběcí stroje, protože jsou nejrozšířenější. CNC je zkratkou anglického „Computer Numerical Control“, pokud tedy mluvíme o obráběcích strojích, můžeme tuto zkratku přeložit jako "počítačem řízený obráběcí stroj". Jedná se o stroje obsahující řídicí systém, který obdrží data ve formě programu a ty postupně převádí na povely jednotlivým částem daného systému, jako jsou například servomotory. Příkazy programu obsahují geometrické, technologické a pomocné informace. Geometrické informace popisují vzájemnou dráhu obrobku a obráběcího nástroje. Technologické informace představují příkazy nezbytné pro dané obrábění, například směr otáčení vřetena, jeho vypnutí nebo výměnu nástroje. Pomocné informace ovládají přidružené funkce stroje jako například chlazení.

Zavedení CNC (NC) strojů mělo za následek zvýšení rychlosti, přesnosti, bezpečnosti práce, lepší podmínky na pracovišti, eliminaci chyb lidského faktoru a také možnost uložení programu pro pozdější nebo opakované využití. Pokud mluvíme o zvýšení rychlosti obrábění, nemáme na mysli jen vyšší rychlost posuvu a hlubší záběr nástroje, ale také zkrácení doby přechodu mezi jednotlivými operacemi. Významnou úsporu času tyto stroje dosáhly i díky své schopnosti automatické výměny nástroje a jeho kalibrování. Přesnost stroje se zvýšila zavedením zpětné vazby a následnou regulací pohonů. Protože při obrábění není potřeba zásahu člověka, obráběcí prostor je po tuto dobu bezpečně uzavřen a chrání obsluhu proti všem případným zraněním. S absencí zásahu člověka do procesu obrábění se obsluha rozdělila na dvě úrovně. První „programátorská“ se stará o vytvoření programu z výkresu součástí. V dnešní době programové vybavení vyspělejších rýsovacích prostředí umožňuje automatické generování programu přímo z vytvořeného výkresu. Druhá „operátorská“ se zabývá přímo obsluhou konkrétního stroje, jejím úkolem je výměna a upnutí materiálu, příprava nástrojů do zásobníku a jejich následné přihlášení do systému, kontrola a případná korekce chyb. Díky pokročilé automatické kontrole jednotlivých systémů se obsluha o případné poruše dozví ještě před vykonáním programu a může zabránit špatnému vyrobení součásti.

### 2.1 Rozdělení CNC

Základní rozdělení CNC obráběcích strojů se provádí podle jejich konstrukce a způsobu obrábění na CNC soustruhy, CNC frézky a CNC obráběcí centra.

### **2.1.1 CNC soustruhy**

Způsob obrábění se od klasických soustruhů výrazně neliší, do vřetene je upnut obrobek a následně roztočen, kolmo na osu rotace se přibližují nástroje, které rotující materiál obrábí. Tyto soustruhy obsahují jedno nebo více vřeten a otočnou hlavici s různými nástroji. Protože při výměně nástroje se otočná hlavice pouze pootočí, dosahují tyto soustruhy rychlosti desetiny až jednotky sekund pro změnu nástroje. Často se setkáme i se soustruhy s možností polohování vřetene a následným frézováním.

### **2.1.2 CNC frézky**

Při frézování se obráběný materiál upne na podložku stolu a z různých směrů se k němu přibližuje rotující nástroj zvaný fréza, který tzv. třískovým obráběním odebírá části materiálu z obrobku. Souřadnicový systém se skládá ze tří os X, Y a Z, jejich směry si vysvětlíme později. Frézky se používají pro výrobu součástí, které nejsou osově souměrné a nedají se obrábět na soustruhu.

### **2.1.3 CNC obráběcí centra**

Obráběcí centra vznikla pro obrábění součástí složitých tvarů. Často v sobě integrují možnosti soustružení a frézování, to vede k zrychlení procesu a ušetření financí za nákup dalšího stroje. Tato centra rozšiřují základní osy (X, Y, Z) o další osy (posuvné i rotační) pro obrábění v nepřístupných místech a pod různými úhly.

## **2.2 Programování CNC strojů [2],[3]**

V dnešní době lze CNC stroje programovat v mnoha různých prostředích a jazycích. V pokročilejších technologiích se programování provádí automaticky přímo z narýsovaného výkresu dané součásti. Ovšem na začátku všeho byl vyvinut a standardizován G-kód. G-kód (ISO-kód) je název programovacího jazyka, který řídí CNC obráběcí stroje. Byl vyvinut společností EIA (Electronic Industries Alliance) počátkem šedesátých let a standardizován v roce 1980 jako RS274D (normy DIN 66025 / ISO 6983). Z důvodů rychlého rozvoje se neustále zvyšovaly nároky na tento programovací jazyk, a tak se většina výrobců odchyluje od tohoto standardu. Někteří výrobci se pokoušeli o zpětnou kompatibilitu s tímto standardem, ovšem nikdy se nepodařilo dosáhnout plné kompatibility při zachování všech funkcí a výhod daného systému. Proto se dnes výrobci snaží odpoutat programátora od G-kódu a nabízejí vyšší programovací jazyky. Pro lepší pochopení bych G-kód přirovnal k Assembleru, který je na rozdíl od vyšších programovacích jazyků např. C závislý na hardwaru.

### 2.2.1 G-kód

Jak již jsem zmiňoval v předchozím odstavci, G-kód je název programovacího jazyka pro CNC stroje. Pomocí tohoto jazyka se programuje dráha nástroje vůči obrobku. Na výběr máme ze dvou režimů polohování - absolutní a relativní. Absolutní polohování udává souřadnice k pevně danému referenčnímu bodu, polohování se tak stává přehlednější. Naopak relativní (přírůstkové) udává souřadnice vzhledem k předchozímu bodu, to sice snižuje přehlednost polohování, ale na druhou stranu se program snadněji vytváří. Tabulka č. 2.1 obsahuje přehled nejpoužívanějších G-kódů.

Funkce	Význam	Adresy
<b>G00</b>	RYCHLOPOSUV	<b>X Y Z</b>
<b>G01</b>	LINEÁRNÍ INTERPOLACE	<b>X Y Z F</b>
<b>G02</b>	KRUHOVÁ INTERPOLACE ve směru hodinových ručiček	<b>X Y Z R F</b>
<b>G03</b>	KRUHOVÁ INTERPOLACE proti směru hodinových ručiček	<b>X Y Z R F</b>
<b>G04</b>	ČASOVÁ PRODLEVA – přerušení běhu programu na dobu (T) sekund	<b>T</b>
<b>G25</b>	SKOK DO PODPROGRAMU – skok do podprogramu na adrese (L)	<b>L</b>
<b>G26</b>	PROGRAMOVÝ CYKLUS – skok do podprogramu na adrese (L) s počtem opakování uvedeným na adrese (H)	<b>L H</b>
<b>G27</b>	PROGRAMOVÝ SKOK – skok na libovolný blok programu uvedený na adrese (L)	<b>L</b>
<b>G29</b>	TEXTOVÁ POZNÁMKA – text v rozsahu dvaceti znaků v ACII	
<b>G31</b>	NAJETÍ NA SONDU v jedné ose	<b>X Y Z</b>
<b>G40</b>	ZRUŠENÍ KOREKCE	

**Tabulka č. 2.1 – G-kód [2]**

### 2.2.2 M-kód

M-kód se využívá pro řízení ostatních náležitostí při obrábění, například automatickou výměnu nástroje, směr rotace nástroje nebo ovládání chlazení. Používá se i pro řízení programových cyklů, přerušení nebo ukončení programu. Lze pomocí něj ovládat i přidružené systémy jako například upínání obrobku. Tabulka č. 2.2 obsahuje vybrané M-kódy.

<b>Funkce</b>	<b>Význam</b>	<b>Adresy</b>
<b>M00</b>	PROGRAMOVÉ ZASTAVENÍ – zastavení běhu programu a otáček vřetena	
<b>M02</b>	KONEC PROGRAMU	
<b>M03</b>	ROZTOČENÍ VŘETENE DOPRAVA – roztočení vřetena v kladném směru na hodnotu otáček (S)	<b>S</b>
<b>M04</b>	ROZTOČENÍ VŘETENE DOLEVA – roztočení vřetena v záporném směru na hodnotu otáček (S)	<b>S</b>
<b>M05</b>	ZASTAVENÍ OTÁČEK VŘETENA	
<b>M06</b>	VÝMĚNA NÁSTROJE – zastavení otáček vřetena a výměna nástroje číslo (T), průměr (D), délková korekce (Z)	<b>T D Z</b>
<b>M07</b>	ZAPNUTÍ CHLADICÍ KAPALINY	
<b>M09</b>	VYPNUTÍ CHLADICÍ KAPALINY	
<b>M17</b>	KONEC PODPROGRAMU	
<b>M99</b>	DEFINICE VELIKOSTI POSUVU	

**Tabulka č. 2.2 – M-kód [2]**

## **2.3 CNC Frézky**

Mým úkolem v této práci je přestavba modelu manipulátoru na CNC frézku, proto se budu dále věnovat jen frézám. Pokuším se popsat jednotlivé části a jejich funkce tak, jak opravdu v praxi fungují a jak jsou vyrobeny.

### **2.3.1 Konstrukce [5]**

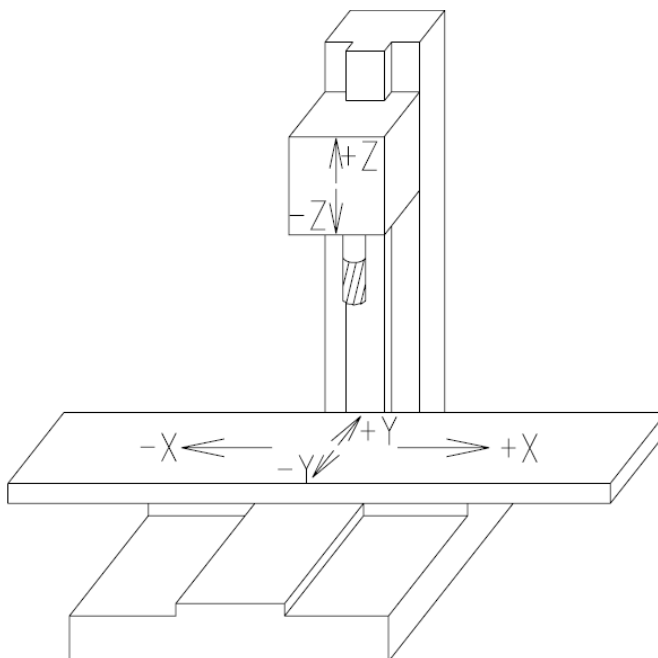
Podle konstrukce dělíme frézky na dva základní typy, a to horizontální a vertikální. Rozlišení těchto dvou druhů spočívá v ose rotace hlavního vřetena, pokud tedy hlavní vřeteno rotuje kolem svislé osy, nazývá se tato frézka vertikální, pokud kolem vodorovné osy, jedná se o horizontální frézku. Rozdíl mezi jednotlivými typy spočívá v poloze upnutí obrobku a směru, ze kterého se nástroj přibližuje (viz. obr. č. 2.1). V praxi se častěji setkáme s vertikálním provedením, ale horizontální provedení nejsou výjimkou.



**Obrázek č. 2.1 - Vertikální a horizontální vřeteno [5]**

### **2.3.2 Souřadný systém**

Hlavním parametrem souřadného systému je počet os. Základní souřadný systém pro frézku obsahuje tři osy: X, Y a Z, rozmístěné podle obrázku č. 2.2. Taková kombinace os nám umožňuje pohybovat se kdekoli v prostoru, ovšem bez možnosti náklonu nástroje a obrábění těžko dostupných míst. V takových případech se do souřadného systému přidávají další dvě osy A a B, tyto osy jsou rotační a umožňují náklon nástroje a jeho nasměrování na libovolný směr (viz. Obr. č. 2.3). Další parametr související se souřadným systémem se nazývá rozjezd os a určuje prostor, ve kterém se lze s nástrojem pohybovat. Tento parametr výrazně ovlivňuje maximální velikost obrobku.



**Obrázek č. 2.2 – Souřadný systém [2]**



Obrázek č. 2.3 – Osy A a B [6]

### 2.3.3 Přesnost

Přesnost se v dnešní době stává parametrem, se kterým se výrobci prezentují a soutěží o nejpreciznější stroj. Přesnost při frézování se klade především na polohování, protože tato přesnost určuje tolerované rozměry obráběné součásti. Přesnost řízení otáček rotace vřetene není při přesném obrábění klíčová. V praxi se odměřování polohy jednotlivých os provádí jednak na snímači motoru, ale také přímo na jednotlivých osách. Odměřování pomocí posuvných snímačů na osách v dnešní době dosahuje přesnosti řádu tisícín milimetru a zároveň nejsou ovlivňovány vůlemi v převodech, proto se považují za přesnější. Snímání polohy se provádí dvojnásobně, aby bylo možné porovnáním obou hodnot zjistit míru opotřebení převodů a varovat obsluhu při překročení tolerance.

### 2.3.4 Nástroje

Požadavky a složitost obrábění se pořád zvyšují, a proto je nutné rozšiřovat i nástrojovou sadu pro obráběcí stroje. Zásobníky dnešních frézek pojmu většinou 10, 30 nebo 50 nástrojů. Tyto nástroje se přihlásí do systému stroje, aby se mohla provádět automatická výměna při obrábění. Pokud řídicí systém narazí při vykonávání uživatelského programu na příkaz výměny nástroje, pozastaví obrábění, najede na pozici, kde se nachází zásobník, který se nastaví na prázdné místo. Zde se nástroj uvolní, zásobník provede výměnu nástroje a nový nástroj je znovu uchycen do vřetena. Následně může být provedena kalibrace nástroje pomocí laseru nebo dotykové sondy. Celý tento proces trvá řádově jednotky sekund, a proto se při automatické výměně ušetří čas vedoucí ke zkrácení doby obrábění. Zásobníky a kalibrovací zařízení se po dobu obrábění chrání krytem proti vniknutí nečistot. Samotné nástroje mohou dosahovat nejrozumnějších tvarů a rozměrů potřebných k jejich specializaci, například hrubé obrábění nebo jemné přesné doladění rozměrů. Z důvodů rychlejší výměny opotřebovaného nástroje za nový se používají nástroje s vyměnitelnými břity (obr. č. 2.4). Výměna pak probíhá



pouhým otočením nebo výměnou břitu, který se přichytí k nástroji šroubem. Pro náročnější aplikace lze volit nástroje umožňující vnitřní chlazení, u kterých probíhá chlazení vstřikováním chladicí kapaliny středem dutého nástroje.



Obrázek č. 2.4 – Výměnné břity [10]

### 2.3.5 Ostatní parametry

CNC frézky se neustále rozvíjejí, jedná se o složité a komplexní zařízení, a proto obsahují nespočet parametrů. Při výběru takového stroje si musí zákazník rozmyslet, jaké nároky budou na daný stroj kladeny a rozhodovat se podle nich. Hlavní parametry jsem již popsal v předchozích odstavcích, a tak další parametry uvedu již jen stručně v krátkém souhrnu. Při výběru CNC stroje nás zajímají i vnější rozměry a váha stroje, aby bylo možno stroj nainstalovat do zvoleného prostoru. Vnitřní rozměry určují maximální velikost obrobku a mohou se tak stát jedním z podstatných parametrů. CNC frézky při své činnosti často používají hydraulická a pneumatická zařízení a je nutné volit tyto součásti s ohledem na již instalované systémy, aby nedocházelo k jejich zbytečnému zdvojení. Možnosti systému chlazení souvisí s možnostmi obrábění, musíme tedy volit podle našich požadavků. Mnoho firem využívá již některá jiná CNC zařízení, jejich programátoři a obsluha pracují již s nějakým řídicím systémem, na který si již zvykli a byli vyškoleni nebo preferují nejvíce vyhovující systém pro jejich potřeby či programové vybavení. Z tohoto hlediska se jeví jako vhodné vybírat i podle tohoto parametru. Požadované napájecí napětí a potřebný proud by v určitých případech mohl zabránit nebo prodražit instalaci stroje. Instalaci a záruční údržbu provádí zpravidla dodavatelská firma a zákazník tak nepotřebuje najímat odborníky v tomto oboru. Strojírenský průmysl se v České republice poměrně rozšířil a není tak problém s dostupností náhradních dílů a nástrojů. Průměrná životnost CNC strojů se pohybuje kolem 5 let, poté se z důvodů opotřebení a ztráty přesnosti doporučuje generální oprava.

### 3 POPIS MANIPULÁTORU [12], [13]

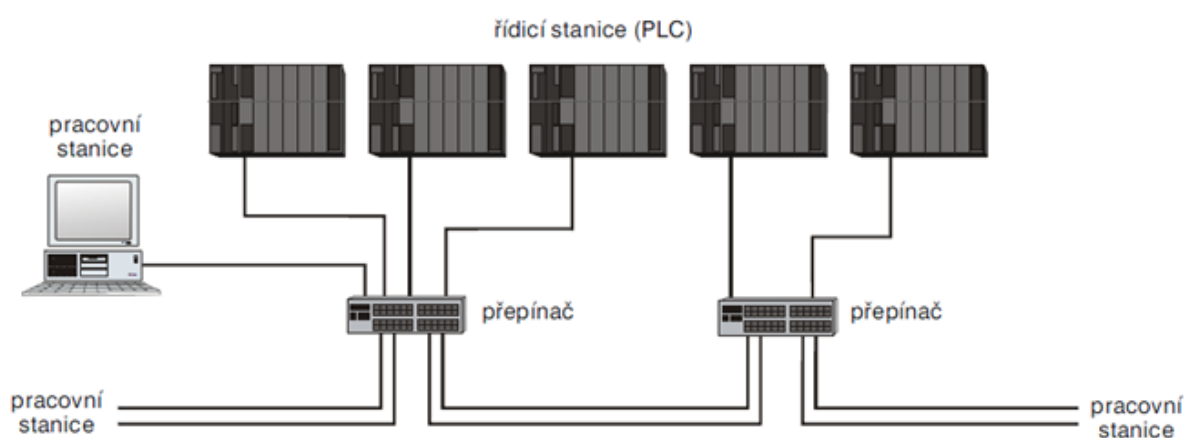
Model manipulátoru v laboratoři E132 byl vyroben firmami Robotech a Zámečnictví – MSK, jeho přístrojové vybavení doplnil Bc. Štefan Golis v rámci své bakalářské práce s názvem Řízení modelu manipulátoru. Jednotlivé systémy budou popsány v následující kapitole, ale nyní si popíšeme manipulátor jako celek. Manipulátor disponuje celkem třemi osami: X, Y a Z. Osy X a Y tvoří souřadnou síť na povrchu stolu a osa Z určuje výšku nad stolem.



**Obrázek č. 3.1 – Model manipulátoru**

Řídicí systém a pohony dodala firma Allen Bradley. Skládá se z PLC ControlLogix obsahující procesory z řady GuardLogix, ethernetovou komunikační kartu, kartu pro komunikaci po síti DeviceNet a interface pro komunikaci se servoměniči Sercos. Systém řízení servomotorů se nazývá Kinetix 2000 a samotné servomotory jsou z řady MPL, vše od firmy Allen Bradley. PLC a servoměniče Kinetix 2000 komunikují pomocí optického rozhraní Sercos. Pro bezpečný chod a ochranu před úrazem se na modelu nachází také bezpečnostní distribuované vstupy/výstupy, na které je připojena optická závora v přední části manipulátoru a také nouzové vypínače. Bezpečnostní distribuovaný I/O modul využívá komunikační síť DeviceNet.

V popisu ethernetové karty se dozvíme o možnosti použít pro komunikaci „obyčejnou“ ethernetovou kartu v PC, čehož je využito v laboratoři. Pokud k tomu přidáme switch, který je ovšem z důvodů časově náročné odezvy již specializován pro průmyslové použití, můžeme vytvořit síť, ve které se nacházejí jak laboratorní PC i jednotlivé PLC systémy (obr. č. 3.2). Modul Ethernet/IP lze v systému použít pro pohodlnou konfiguraci PLC a ostatních částí řídicího systému, jeho programování a sledování jeho stavu přímo z PC. PLC sestava použitá pro řízení našeho modelu neobsahuje modul digitálních vstupů/výstupů, a proto se pomocí této sítě můžeme připojit k ostatním PLC, které tyto moduly obsahují. Síť Ethernet/IP můžeme využít ještě pro vizualizaci s některou z jednotek PanelView.



Obrázek č. 3.2 – Schéma sítě Ethernet/IP [11]

### 3.1 PLC ControlLogix [15]

Jedná se o modulární systém od firmy Allen Bradley [35][36][37]. Jednotlivé moduly se umísťují do šasi s napájecím zdrojem, šasi se vyrábí v různých velikostech pro 4, 7, 10, 13 a 17 modulů. Umístění modulů v šasi je libovolné, může zde být i více procesorů, v takovém případě si každý procesor po nahrání svého programu vyhledá moduly, které bude spravovat a nahraje do nich konfigurační data, tato data určují způsob a periodu komunikace. Systém umožňuje vkládání a vyjímání modulů nebo jejich svorkovnic za chodu automatu. V případě potřeby většího počtu modulů se mohou šasi propojit některou z komunikačních jednotek, nejčastěji sítí ControlNet. Pro vnitřní komunikaci využívá sběrnici ControlBus. Tato sběrnice využívá kromě komunikace klient/server pro přenos časově nekritických dat také metodu producent/spotřebitel pro přenos cyklických časově kritických dat. Výhodou je i možnost opatření dat časovou značkou. Všechny moduly sdílí stejná přístupová práva ke sběrnici.

### 3.2 Procesory 1756-L62S a 1756-LSP [16]

Procesor GuardLogix je procesor ControlLogix, který navíc poskytuje bezpečnostní kontrolu řízeného systému. Systém GuardLogix je duální procesor, proto se vždy musí použít primární procesor 1756-L6xS a sekundární safety 1756-LSP (obr. č. 3.3). GuardLogix splňuje bezpečnostní úroveň SIL 3 (Safety integrity level) nebo CAT.4. Hlavní výhodou tohoto systému spočívá v tom, že při programování tohoto systému vytváříme pouze jeden projekt, který obsahuje již oba tyto procesory, čímž se zjednoduší konfigurace projektu a následné programování.

Jedná se tedy o procesory z řady GuardLogix 1756-L6xS a 1756-LSP s uživatelskou pamětí 4MB (1MB safety), pamětí CompactFlash o velikosti 64MB pro zálohování programu v době vypnutí PLC. Součástí procesoru je i hardwarová a softwarová diagnostika. Moduly lze programovat pomocí integrovaného rozhraní RS-232 nebo pomocí jiného komunikačního modulu (Ethernet/IP, ControlNet, DeviceNet atd.) v šasi. Paměť vyhrazená pro vstupy a výstupy má velikost 478MB, a proto lze připojit až 128 000 digitálních a 4000 analogových vstupů a výstupů. Modul ke své práci nepotřebuje zálohovací paměť CompactFlash ani lithiovou baterii, ovšem v takovémto případě nedojde k zálohování a znovunačtení dat ze zálohovací paměti a PLC si po vypnutí a opětovném zapnutí nebude nic pamatovat. Procesor lze programovat reléovým diagramem (LD), strukturovaným textem, funkčními bloky (FB) anebo sekvenčními funkčními diagramy (SFC).



Obrázek č. 3.3 – Procesory PLC [17]

### 3.3 Ethernet/IP modul [18]

Síť EtherNet/IP (EtherNet/Industrial Protocol) byla standardizována v roce 2005 pod názvem IEC 62413. Mezi její hlavní výhody patří plná kompatibilita se sítěmi Ethernet, v případě potřeby lze tedy použít klasické komponenty Ethernetu, například k připojení PC do sítě Ethernet/IP postačí pouze „obyčejná“ síťová karta, což je velmi levné, rychlé a dostupné řešení. Bezpečnost a reálnou odezvu tedy zaručuje aplikační vrstva tvořená protokolem CIP (Common Industrial Protocol), který se využívá i u sítí ControlNet a DeviceNet. Síť Ethernet/IP [40] používá dva druhy přenosu: explicitní (TCP/IP) přenos typu žádost - odpověď mezi dvěma uzly a implicitní (UDP/IP) pro cyklický přenos uživatelských a I/O dat. Protokol CIP tak používá dva principy komunikace: nespojovaná pro přenos I/O dat s vysokou prioritou a spojovaná pro přenos nepravidelných dat s nízkou prioritou. Reálná odezva je v této síti zajištěna rozdělením kolizních domén pomocí switchů a dostatečnou kapacitou sítě. Pro náročnější aplikace se využívá mechanismus distribuovaných hodin (CIP Sync) a kompenzace doby přenosu.



Obrázek č. 3.4 – Ethernet/IP modul [19]

### 3.4 DeviceNet modul [20],[21]

Síť DeviceNet byla vyvinuta americkou firmou Allen-Bradley (nyní Rockwell Automation). Aby se síť DeviceNet rychleji rozšířila, uvolnila firma Rockwell Automation tento standard. Nyní organizace Open DeviceNet Vendors Association (ODVA) dohlíží na specifikace DeviceNet a zajišťuje testování shody se standardy DeviceNet. Síť DeviceNet je založena na sítích CAN (Controller Area Network), který byl původně vyvinut pro automobilový průmysl. Specifikace CAN nedefinuje aplikační vrstvu a částečně fyzickou vrstvu, a proto síť DeviceNet definuje pouze tyto dvě vrstvy. Přenosové rychlosti této sítě jsou 125, 250 a 500Kbps, maximální délka metalického vedení 500m a připojit se může až 64 zařízení (uzlové adresy v rozsahu 0 - 63). Délka sběrnice je nepřímo úměrná rychlosti, tj. 500, 250 a 100 metrů. Přenosové vedení se skládá ze dvou vodičů napájení, dvou datových vodičů a stínění. Podporuje master/slave i peer-to-peer komunikaci (častěji master/slave). Standard dovoluje více masterů na jednu logickou síť. Kabel DeviceNet může napájet méně náročné zařízení, jako jsou koncové spínače nebo snímače. Síť byla navržena pro provoz v zarušeném prostředí (odolnější než RS-485).

#### Fyzická vrstva

Uzly jsou distribuovány po páteřní síti DeviceNet pomocí průchozích konektorů. Tato topologie umožňuje snadné a rychlé připojení nebo odpojení uzlu. To má za následek zvýšení flexibility sítě, řešení problémů a snížení montážního času. Fyzická vrstva je od ostatních vrstev opticky oddělena, čímž se zabrání poškození zařízení při zkratu na sběrnici a zvyšuje bezpečnost i možnost napájení po stejném kabelu.



Obrázek č. 3.5 – DeviceNet modul [22]

### 3.5 SERCOS interface [14],[23]

Sercos (SErial Real-time COmmunications System) je otevřený komunikační protokol pro řízení servopohonů Allen Bradley Kinetix 6000, Ultra 3000, 8720MC a 1394. Specializuje se na řízení digitálních pohonů, kde je kladen důraz na komunikaci v reálném čase a díky optickému rozhraní se výborně hodí do zarušených prostředí. Systém Sercos je určen ke spolupráci s pohony Allen Bradley a moduly ControlLogix. Modul SERCOS 1756-M08SE je určen pro řízení systému až s osmi osami. Nespornou výhodou tohoto systému je jisté použití optických vláken, z čehož vyplývá vysoká odolnost proti el. mag. rušení a možnost řízení na velké vzdálenosti (až 32 m při použití plastických vláken a až 200 m při použití vláken skleněných). Uživatelsky příjemná je také možnost konfigurace a monitorování pomocí standardního rozhraní RSLogix 5000, ve kterém je zde možno využít až 32 instrukcí. Další výhodou spočívá v možnosti připojení zařízení za chodu a jeho výměna bez konfigurace. V systému je zabudovaná i diagnostika, která informuje řídicí systém o stavu pohonu.

#### Přehled základních parametrů:

Max počet serv:	8
Rychlost přenosu:	4 nebo 8 Mbps
Cyklus opakování dat:	0,5ms (2 serva) 4Mbps, 0,5ms (4 serva) 8Mbps 1ms (4 serva) 4Mbps, 1ms (8 serva) 8Mbps 2ms (8 serv) 4Mbps, 2ms (16 serva) 8Mbps
Řízení:	Rychlost, poloha, moment
Ztrátový výkon:	5W
Spotřeba:	760mA 5V (2.5mA 24V)
Hmotnost:	30g
Izolační pevnost:	4kV
Operační teplota:	max. trvalá teplota 60°C
Katalogové číslo:	1756-M08SE



### 3.6 Kinetix 2000 [24],[25]

Jedná se o nejslabší systém pro řízení více spolu spjatých servomotoů od firmy Allen Bradley z řady Kinetix. Maximální přípustný výkon připojeného motoru dosahuje 3kW. Z toho vyplývá, že se jedná o systém určený pro jemné aplikace nenáročné na výkony servomotorů. Systém komunikuje s již zmíněným rozhraním Sercos pomocí optických kabelů. Systém je optimalizován pro řídicí systémy ControlLogix a jeho konfiguraci lze provést přímo z prostředí RSLogix5000. Napájení měničů lze volit jedno nebo třífázové s rozsahem napětí 100 až 230V. Kinetix 2000 byl vyvinut a optimalizován pro řady motorů MP a TL (Allen Bradley). Systém využívá modulární konfigurace. První servoměnič obsahuje komunikační interface Sercos a také napájecí zdroj, pomocí kterého se napájí ostatní měniče, které přidáváme do následujících slotů. Systém Kinetix 2000 může obsahovat až 8 servoměničů napájených z jednoho zdroje. Použitý modul s napájecím zdrojem a komunikačním interfacem nese název Kinetix 2000 2093-AC05-MP5 a ostatní moduly Kinetix 2000 2093-AMP5.

#### Přehled základních parametrů:

Max. počet serv:	8
Napájecí napětí:	170 - 264VAC, 1f nebo 3f
Frekvence:	50 - 60Hz
Vstupní proud:	11.6A (3f, 230V), 10.95A (1f, 230V)
Výstupní proud:	9.67A (3f, 230V), 6.42A (1f, 230V)
Výstupní výkon:	3kW (3f, 230V), 2kW (1f, 230V)
Operační teplota:	max. trvalá teplota 60°C
Katalogové číslo:	1756-M08SE



Obrázek č. 3.6 – Kinetix 2000 [24]



### 3.7 Servomotory [26],[27]

Servomotory řady MPL (MP Low Inertia Servo Motors) se vyznačují nízkou setrvačností, způsobenou jejich sníženou hmotností. Jedná se o střídavé bezkartáčové motory se zpětnou vazbou, kterou výrobce nabízí ve variantách absolutní, inkrementální a resolverové. Tělo motoru je vyrobeno z nerezové oceli, aby se motor mohl používat i v potravinářském průmyslu a hřídel je kalená pro přenos vyššího momentu. Motory byly vyvinuty pro plnou kompatibilitu se systémy Allen-Bradley Kinetix 2000, Kinetix 6000, Kinetix 7000 a Ultra3000. Výrobce nabízí možnost integrovat 24V stejnosměrnou brzdu pro zabrzdění hřídele v době nečinnosti motoru. Přívodní kabely silové i datové přehledně a jednoduše připojujeme pomocí konektoru, což nám znemožní jejich záměnu a usnadní zapojení. V řadě MPL se nachází motory s napětím 230V i 460V s otáčkami až 5000 ot/min a krátkodobým momentem až 278Nm. Motory použité v naší úloze jsou napájeny napětím 320V, jejich maximální otáčky dosahují 3000 ot/min a jsou vybaveny inkrementální zpětnou vazbou. Motor v ose Z je navíc vybaven již zmiňovanou 24VDC brzdou.

#### **Přehled základních parametrů:**

Max otáčky:	3000 ot/min
Napájecí napětí:	230 VAC, 3f
Frekvence:	50 - 60Hz
Moment:	1.58 Nm
Hmotnost:	3 Kg
DC Brzda:	24VDC (pouze osa Z)
Zpětná vazba:	inkrementální enkodér
Operační teplota:	max. trvalá teplota 40°C
Krytí:	IP50
Katalogové číslo:	MPL A310 HK24AA, A310F HK22AA



**Obrázek č. 3.7 – MPL A310 [27]**

## 4 ÚPRAVY MANIPULÁTORU

Protože se model manipulátoru má stát CNC frézkou, je nutné některé jeho části přizpůsobit tomuto účelu. Během semestrálních projektů jsem se zamýšlel, jaká bude další činnost manipulátoru a následně nad možnými riziky a požadavky pro práci v laboratoři. Při navrhování úprav jsem se zaměřil jednak na nejdůležitější úpravy nutné k chodu stroje, jeho zabezpečení proti úrazu a následně k možnostem, které sebou přináší finanční rozpočet a provoz v laboratoři. Při zvážení všech požadavků a omezení jsem navrhnul následující úpravy.

### 4.1 Ochranné kryty

Frézování se řadí mezi třískové obrábění, kdy nástroj vyřezává malé části materiálu z obráběného předmětu. Tyto odřezané části jsou následně rotací nástroje vrženy ve směru tečny a jeho rotace. Vzhledem k tomu, že se jedná o ostré části vržené určitou rychlostí, nesmí se dostat do kontaktu s lidmi nebo předměty, které by mohly poškodit. Pro ochranu osob, manipulátoru a také ostatního vybavení laboratoře je nutné manipulátor vybavit kryty, které zamezí vylétávání nežádoucích částí materiálu z prostoru obrábění.

Během mé práce jsme se kryty nezabývali a jejich konstrukci jsme jen zběžně probrali a odložili na dobu, až nebude manipulátor využíván ostatními kolegy při jejich závěrečných pracích. Předběžná domluva počítá s namontováním krytu, který by měl být ze tří stran neodnímatelně upevněn, a z přední strany by měly být namontovány dvířka pro přístup do obráběcího prostoru. Pro výrobu krytů navrhuji použít plexisklo s dostatečnou pevností a šířkou (asi 5mm).

### 4.2 Stůl na upnutí

Protože při obrábění na sebe nástroj a obrobek působí určitou silou, je nezbytné upevnit obráběný předmět na obráběcím stole. Stůl nacházející se v manipulátoru je vyroben ze dřeva a nelze k němu upevnit žádný předmět. V praxi se k upnutí obráběného předmětu při frézování používá mechanismus, kdy se do stolu vyfrézují drážky s profilem ve tvaru obráceného písmene „T“ a obrobek se k němu přitiskne přes šrouby, které se zachytí na jedné straně do zmíněné drážky a na druhé se pomocí speciální podložky a matky přitiskne k povrchu stolu. Upínání se provádí vždy více šrouby najednou, aby nemohlo dojít k uvolnění materiálu při obrábění. Další ze způsobů upínání obrobku spočívá v nainstalování svěracího mechanismu, do kterého se následně upne obrobek. Tento způsob sebou nese nevýhodu v podobě omezené velikosti upínaného materiálu, možnosti upnout vždy jen jeden obrobek a pevně dané místo upínání.

V této práci jsme se upínacím stolem prakticky nezabývali. Upínání bude vyřešeno opět až v době uvolnění manipulátoru. Jako doporučení navrhuji použít jiný stůl s klasickým upínáním nebo nainstalovat odnímatelný svěrací mechanismus.

### **4.3 Výměna nástroje**

Pro plnění rozmanitých požadavků při frézování si nevystačíme s jedním řezným nástrojem. Je nutné tyto nástroje střídát. Z důvodů rychlosti a eliminace zásahu obsluhy do obráběcího procesu se tato výměna v CNC strojích provádí plně automaticky. Pro automatickou výměnu nástroje se v dnešních obráběcích systémech nachází držák na nástroje, do kterého obsluha před začátkem obrábění uloží nástroje, které následně navolí v řídicím systému. Pokud při provádění programu narazí řídicí systém na změnu nástroje, přeruší obrábění, zastaví rotaci nástroje, otevře kryt chránící uložené náhradní nástroje, do volné pozice uloží dosud používaný nástroj a vyzvedne si nově zvolený. Tuto funkci musí podporovat mechanismus na upnutí nástroje. V dnešní praxi se upínání nástrojů provádí pomocí hydraulických upínacích hlav. Pro korekci případných nepřesností v polohování najíždí tyto hlavy na kuželovou stopku nástroje. Aby se předešlo chybám způsobených nečistotami v upínacím mechanismu, po každém vyjmutí nástroje následuje krátké vyfouknutí případných nečistot z upínacího mechanismu stlačeným vzduchem.

Protože model manipulátoru se během mého působení přetvoří na gravírovací CNC frézku a ne na plnohodnotnou CNC frézku, není nutné používání mnoha nástrojů. Vzhledem k finančním a technickým možnostem se výměna nástroje bude provádět jen v případě poškození původního nebo jen velmi zřídka. Upínací hlava neumožňuje automatické uvolnění nástroje, a proto výměnu provede obsluha ve spolupráci s řídicím systémem. Pokud řídicí systém zjistí požadavek na výměnu nástroje, pozastaví obrábění, zastaví rotaci nástroje a upínací hlavice se přesune do určené polohy, kde obsluha provede výměnu nástroje. Následně potvrdí dokončení této výměny a systém bude pokračovat v obrábění.

### **4.4 Rotace nástroje**

Jedna z nejdůležitějších úprav manipulátoru proto, aby mohl sloužit jako frézka, spočívá v nainstalování ústrojí pro rotaci a upínání řezného nástroje na část manipulátoru, která se pohybuje ve všech třech osách. V praxi se pro tento pohon využívají motory s výkonem v řádu jednotek až desítek kW, to ale v našem případě nebude možné ani potřebné. V závislosti na omezeném prostoru, do kterého musí být motor namontován, nelze použít převodovku, a tak je nutné motor navrhnout tak, aby splnil všechny požadavky, které jsou nezbytné pro funkčnost zařízení. Jeden z hlavních požadavků kladených na tento pohon je schopnost měnit otáčky ve velkém rozsahu. Přesnost těchto otáček již není tak kritická jako u servomotorů použitých pro polohování. V naší úloze se klade vysoký nárok na přesnost polohování a tento

motor se nachází na samém konci polohovacího řetězce a s největší pravděpodobností zde nebude převodovka, je tedy nutné zajistit co nejmenší vůli v ložiskách tohoto motoru, aby nedocházelo k nepřesnostem. Další důležitou věc, kterou musíme zohlednit při výběru motoru, je také krytí motoru a přívodních kabelů nebo opatření celého ústrojí krytem zabráňujícím poškození v důsledku odlétávajících částí materiálu při obrábění.

Pro vyřešení všech požadavků kladených na tuto část zařízení a s ohledem na finanční možnosti se jevílo jako nejelegantnější a nejjednodušší řešení pořídit běžně prodávanou ruční frézku. Tyto frézky již splňují základní požadavky pro jejich provoz a vyřešily by se tak problémy spojené s krytím, požadovanými otáčkami i upínacím zařízením. V průběhu mé práce pořídil vedoucí práce sadu obsahující ruční frézku s příslušenstvím od firmy Dremel. Samotná frézka (obr. č. 4.1) již řeší nejen motor s jeho výkonem a převodováním ale také jeho uložení spolu s přepínačem rychlostí. Další výhodou je bezesporu i hlavice pro upevnění nástroje a nástroj v podobě miniaturní frézy, který se v sadě také nachází. Předpokládám, že výkon motoru i jeho otáčky jsou navrženy pro přibližně konkrétní rozměry a tvar nástroje.



Obrázek č. 4.1 – Ruční frézka [9]

## 4.5 Upevnění frézky

Při implementaci frézky na náš manipulátor je potřeba vyřešit upevnění ruční frézky na rameno manipulátoru. Pro upevnění frézky se na jejím spodním okraji šasi nachází závit, za který se frézka upevňuje do ručního stojanu. Po zhlédnutí celé její konstrukce jsem se rozhodl využít právě tohoto závitu i pro upevnění do našeho zařízení. Abychom mohli zajistit dostatečně robustní uchycení frézky, bude potřeba vyrobit mezičlánek, který bude možno pomocí šroubů snadno uchytit ke stávající hlavě manipulátoru a kolmo na tuto plochu se bude nacházet závit pro upevnění frézky. Pro

## 4.6 Automatické spouštění

29

## 4.7 Kalibrace nástroje

V praxi se nástroje kalibrují proto, aby se zjistil jejich skutečný poloměr a opotřebení. Kalibrace se provádí najížděním hlavy s upnutým nástrojem na pozici, kde se nachází měřicí systém, který následně vyhodnotí přesnou pozici hrany nástroje a z aktuální pozice středu nástroje (pozice upínací hlavy) se vypočítá jeho poloměr potřebný k přesnému obrábění. Pro velmi přesné určení pozice hrany nástroje se využívá interferenční metoda. Pro méně přesné a levnější varianty kalibrace lze polohu nástroje zjišťovat přerušením laserového paprsku nebo dotykovou sondou, která po dotyku s nástrojem ohlásí tuto událost.

V naší aplikaci nebude potřeba kalibrovat nástroj, protože používané nástroje budou zanedbatelného průměru. A také proto, že se jedná o gravírovací stroj, při jehož práci určujeme střed frézované drážky.

## 4.8 Určování nulových bodů

Pro chod naší frézky budeme potřebovat nulový bod stroje a nulový bod obrobku. Nulový bod stroje se nachází v počátku souřadného systému obráběcího stroje. Většinou se jedná o bod, na který se dostane hlava s nástrojem po zadání kódu „G0 X0 Y0 Z0“, tento bod je považován za referenční pro souřadný systém stroje a řídicího systému. Nulový bod obrobku je posunut vzhledem k nulovému bodu stroje podle toho, kde je upnut obrobek. Jeho určení se v praxi provádí pomocí ručního posuvu, nastavením nástrojové sondy na počáteční hranu obrobku. Toto nastavení se musí provádět pokaždé, když se obrobek vymění nebo se odpíná od stolu. Stroj si následně zapíše hodnoty offsetů os do paměti a poté je při obrábění připočítává k pozici nástroje. Pro naši aplikaci bude dostatečné, pokud si v ručním režimu při zastaveném vřetenu nastavíme pozici, při které se fréza lehce dotýká počátku obrobku, a následně vydáme povel k zapsání offsetů.

## 5 GENEROVÁNÍ G-KÓDU

Pokud bychom g-kód vytvářeli sami, bylo by to krajně nepraktické a podobalo by se to programování procesoru ve zdrojovém kódu. Museli bychom si postupně představovat dráhu nástroje, přepisovat ji do příkazů a přiřazovat k nim souřadnice. Tento postup se v praxi nepoužívá a editování g-kódu se provádí pouze za účelem kontroly nebo úpravy programu. Pokud chceme, aby CNC frézku mohli používat i lidé bez hlubokých znalostí CNC programování, je nutné nabídnout jim cestu, jak snadno a pro ně srozumitelně vytvořit program, který následně nahrají do CNC frézky. Při hledání optimálního způsobu vytváření CNC programu jsem se zaměřil především na srozumitelnost pro osoby používající tento stroj a cenovou dostupnost.

### 5.1 Výběr vhodného softwaru [31]

Po provedení malé studie na internetu se ukázalo, že software tohoto typu je rozmanitý a již dlouho používaný. Nalezneme zde nejrozličnější typy těchto programů od jednoduchých pluginů až po komplexní profesionální nástroje. Profesionální nástroje nabízejí mnoho možností a nastavení, ale v našem případě se jedná spíše o nevýhodu, která daný program dělá složitějším a nutí uživatele studovat velké množství informací. Další nevýhodou profesionálních nástrojů je jejich cena, a proto jsem se vzhledem k těmto překážkám rozhodl využít méně složitější nástroj s volnou licencí. Další řada nástrojů představuje plugin do komplexních nástrojů, jako jsou například AutoCAD. Tyto nástroje jsou mnohdy s volnou licencí, ale nejsou schopny fungovat samostatně a je potřeba mít instalovaný například zmíněný AutoCAD, který již zadarmo není. Použití těchto pluginů je tedy možno, pouze pokud již AutoCAD s licencí vlastníme. Toto omezení však může představovat problém, a tak jsem od tohoto řešení také upustil.

Samostatně fungující software většinou nebyl s vhodnou licencí dostupný, a tak jsem na radu vedoucího zvolil plugin do softwaru EAGLE, který není sice zdarma, ale je mezi studenty i učiteli velice dobře známý, a tak není třeba se učit s tímto programem zacházet. Plugin je původně navržen pro výrobu DPS, kdy se pomocí CNC stroje vyfrézují izolační mezery mezi jednotlivými cestami.

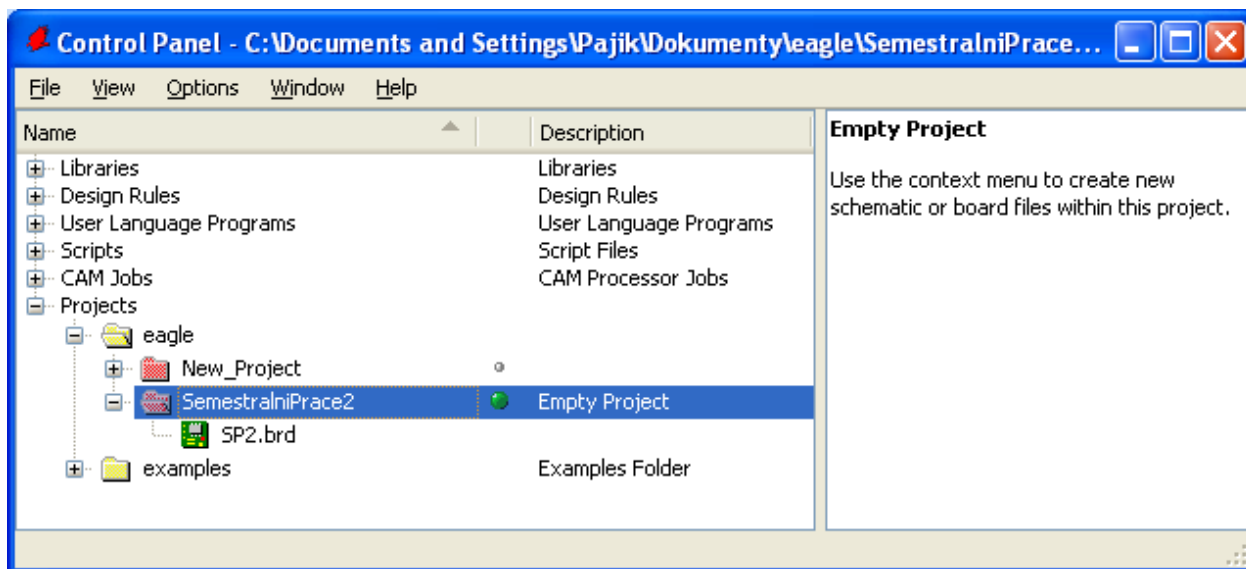
### 5.2 Popis programu PCB-GCODE [32]

Jak jsem již naznačil v předchozím odstavci, jedná se o plug-in do programu Eagle. Tento program vytvořil John Johnson a je šířitelný s licencí Open source. Program byl vyvinut pro generování G-kódu potřebného pro výrobu desky plošných spojů a to jak kódu pro frézování vrchní i spodní strany, tak i pro vrtání otvorů pro nožičky součástek. Pro naše potřeby si program nastavíme tak, aby námi nakreslený

vzor pro gravírování považoval za vytvořenou desku plošných spojů a vytvořil tak soubor s G-kódem pro její výrobu. Jeho instalace se provádí pouhým rozbalením archivu, ve kterém se nachází program a následné přesunutí souborů do složky „UPL“ v kořenovém adresáři programu Eagle. Pro Windows se tato složka typicky nachází na následující adrese: „C:\Program Files\EAGLE-5.6.0\ulp“. Program Eagle podporuje všechny tři základní operační systémy (Windows, Linux i Mac OS) a protože se jedná o plug-in do tohoto programu, i náš program podporuje tyto operační systémy.

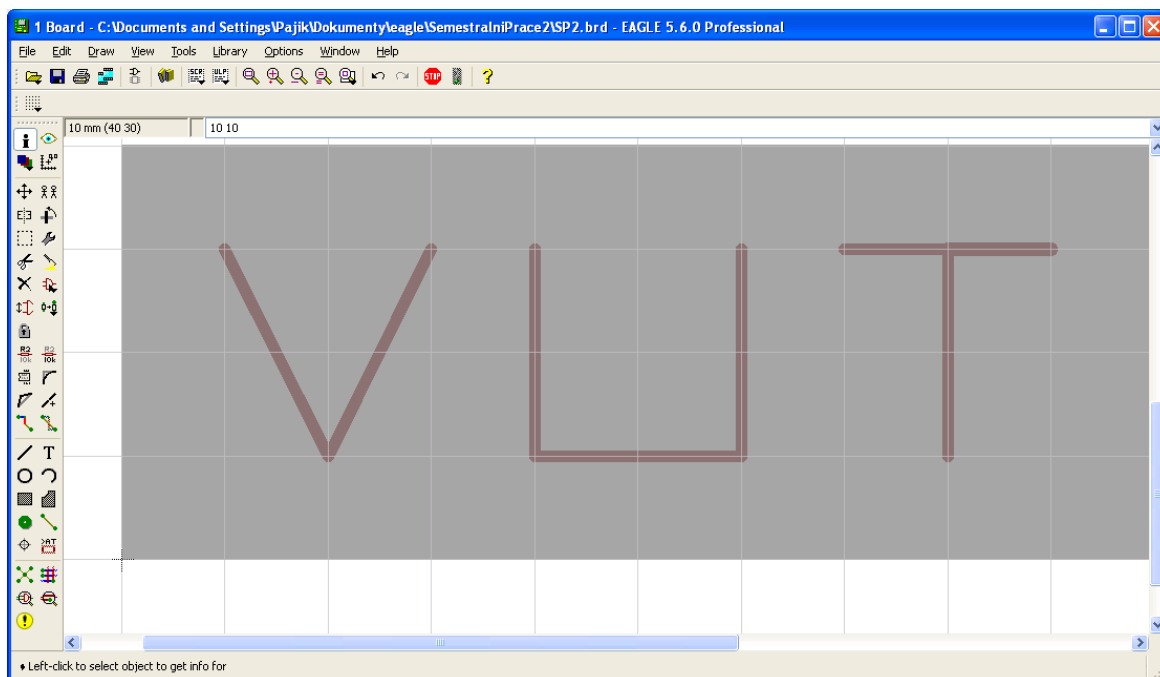
### 5.2.1 Spuštění Eaglu a vytvoření obrazu [33]

Nejprve si spustíme program Eagle a v jeho prostředí (obr. č.: 5.1) si vytvoříme nový projekt, v našem případě „SemestrálníPrace2“. Dále vytvoříme soubor s deskou plošných spojů a uložením ji připojíme do našeho projektu. Celou naši fiktivní desku plošných spojů budeme vytvářet jako jednovrstvou. Nejprve si ohraničíme oblast, na které budeme vytvářet náš vzor pro gravírování. To provedeme vytvořením obdélníku ve vrstvě 21 tPlace. Tento obdélník by měl začínat (levý dolní roh) v počátku souřadného systému, který je označen křížkem v prostoru editoru. Při vytváření si můžeme zapnout mřížku pro lepší orientaci a kontrolu rozměrů. Dále již přistoupíme k návrhu našeho vzoru, ten budeme vytvářet ve vrstvě 1 Top pomocí komponenty „Wire“. Po výběru této komponenty můžeme v horní části editoru zvolit způsob zalomení a šířku vytvořené stopy. Na obrázku č. 5.2 vidíme prostředí editoru desky s vytvořenou plochou a nápisem VUT.

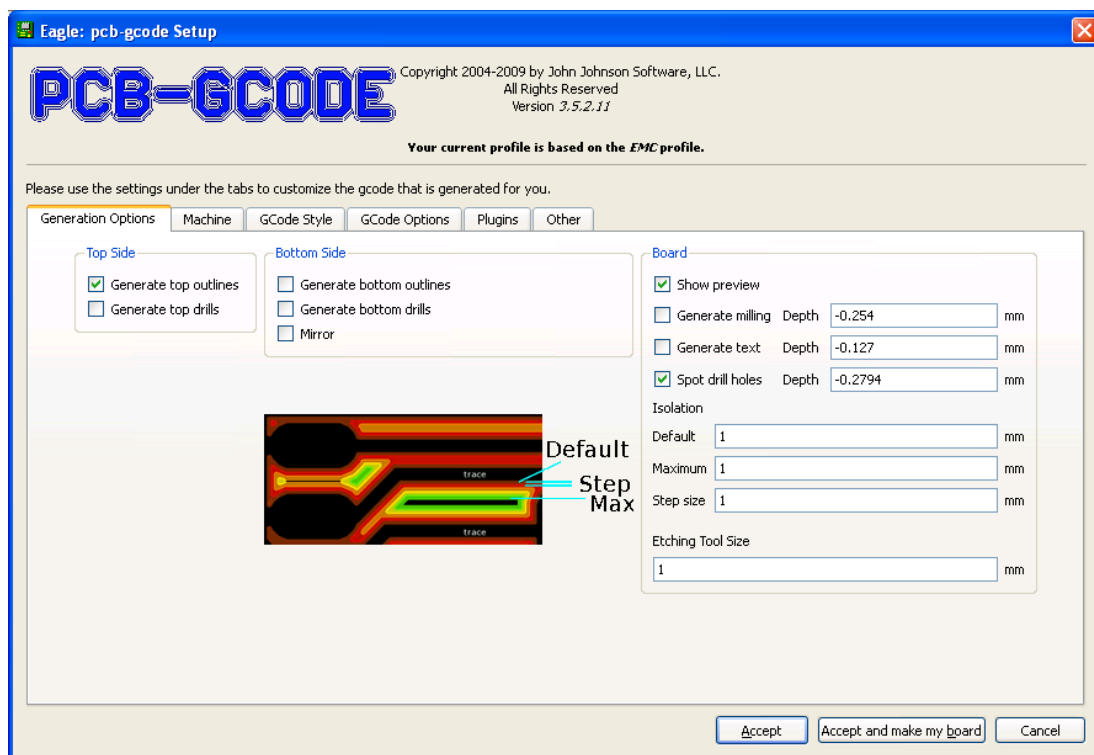


Obrázek č. 5.1 – Prostředí Eagle





Obrázek č. 5.2 – Prostředí editoru desky



Obrázek č. 5.3 – Prostředí PCB GCODE I.

## 5.2.2 Inicializace PCB GCODE

Pro spuštění pluginu „PCB GCODE“ napíšeme do příkazové řádky editoru desky „*run pcb-gcode-setup*“ nebo „*run pcb-gcode*“, při zadání parametru „*-setup*“ se spustí nejprve možná nastavení. Pokud příkaz zadáváme poprvé, nastavení se spustí automaticky. Je možno také využít zkratk Shift+Ctrl+S pro „*run pcb-gcode-setup*“ a Shift+Ctrl+R pro „*run pcb-gcode*“. Po zadání příkazu se spustí okno, které můžeme vidět na obrázku č.: 5.3.

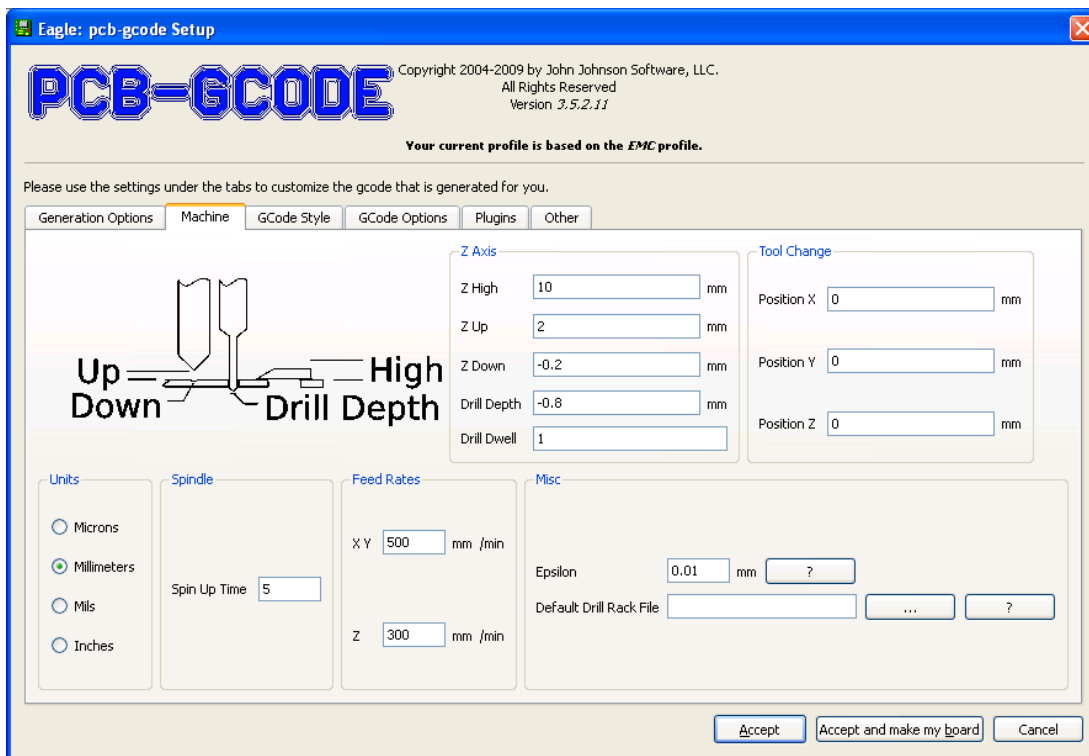
### 5.2.2.1 Generation Options

V části „Top Side“ zatrhneme pouze první možnost „Generate top outlines“ pro vygenerování kódu frézování. Možnost „Generate top drills“ generuje kód pro vrtačku. Část „Bottom Side“ slouží pro nastavení generování kódu spodní strany desky, což v našem případě není potřebné.

V části označené jako „Board“ nastavujeme parametry pro frézování. Záložka „Show preview“ zobrazí náhled frézované oblasti. Dále nastavujeme hloubku frézování dráhy („Generate milling“), textu („Generate text“) a hloubku „důlků“ pro pozdější vrtání. Program je navržen pro vícenásobné frézování, aby se zohlednila maximální hloubka frézování na jeden tah. Hodnota „Default“ označuje hloubku prvního kola frézování, „Maximum“ označuje konečnou hloubku a „Step size“ hloubku jednotlivých frézování. Poslední hodnota „Entlich Tool Size“ určuje šířku použitého nástroje.

### 5.2.2.2 Machine

V této záložce (obr. č. 5.4) pokračujeme v nastavování parametrů frézování. Skupina parametrů „Z Axis“ určuje horizontální meze frézování, vysvětlivky vidíme na obrázku v prostředí. „Z High“ označuje výšku, v jaké se může nástroj volně pohybovat, aniž by došlo ke kolizi s nějakým předmětem, „Z Up“ výšku nástroje pro přemístění při přeskokování vodivých cest, „Z Down“ maximální hloubku frézování, „Drill Depth“ maximální hloubku při vrtání a „Drill Dewell“ čas, po jaký zůstane vrták v dolní poloze. Pozici pro výměnu nástroje určují hodnoty zapsané v „Tool Change“. Jednotky, ve kterých budeme pracovat, volíme přepínačem „Units“. Hodnota „Spindle“ určuje čas nutný k ustálení otáček při zapnutí vřetene. Rychlosti řezných posuvů zapíšeme do kolonek v oblasti „Feed rates“ a nakonec toleranci, při které bude program požadovat dva objekty za jeden, zapíšeme do položky „Epsilon“.



**Obrázek č. 5.4 – Prostředí PCB GCODE II.**

#### 5.2.2.3 GCode Style

Zde vybereme styl generovaného kódu. V našem případě ponecháme první možnost „EMC“.

#### 5.2.2.4 GCode Options

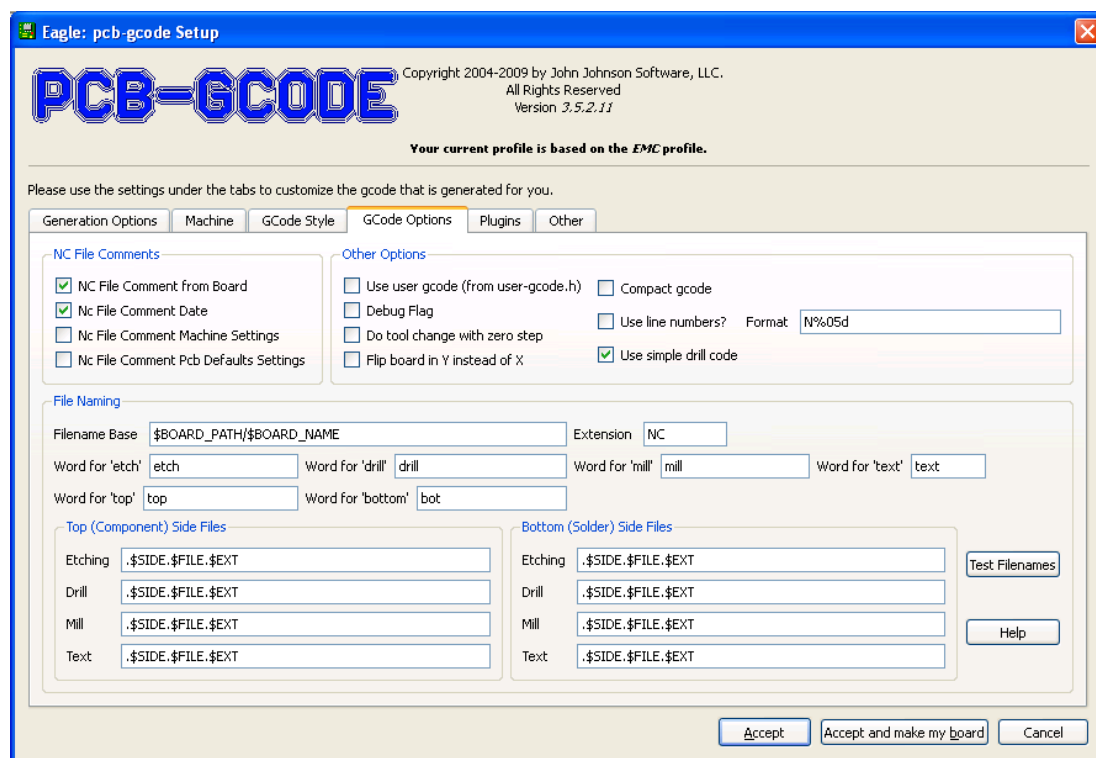
Oblast „NC File Comments“ (obr. č.: 5.5) definuje, co vše se má přidat do souboru s kódem jako komentář. Pro naše potřeby postačí přidat informace, z jakého souboru byl kód vytvořen („NC File Comment from Board“) a základní nastavení frézování („NC File Comment Date“). Oblast „Other Options“ slouží pro nastavení dalších náležitostí kódu. Možnost „Use user code“ zvolíme pro výběr vlastního upraveného souboru tvorby kódu. „Debug flag“ slouží pro případ ladění kódu při jeho chybě. „Do tool change with zero step“ umožňuje při změně nástroje kalibrovat jeho výšku, „Flip board in Y instead of X“ prohození os při otočení desky. „Compact code“ odstraní nadbytečné části příkazů např. G01. „Use line numbers“ přidá do souboru čísla řádků. „Use simple drill code“ pro vrtání použije G01 a G00 namísto G82. Dále jen změníme příponu souboru („Extension“) na NC.

### 5.2.2.5 Plugins

Zde můžeme přidat vytvořené pluginy, defaultně se zde nachází kalkulačka pro potřeby programátora.

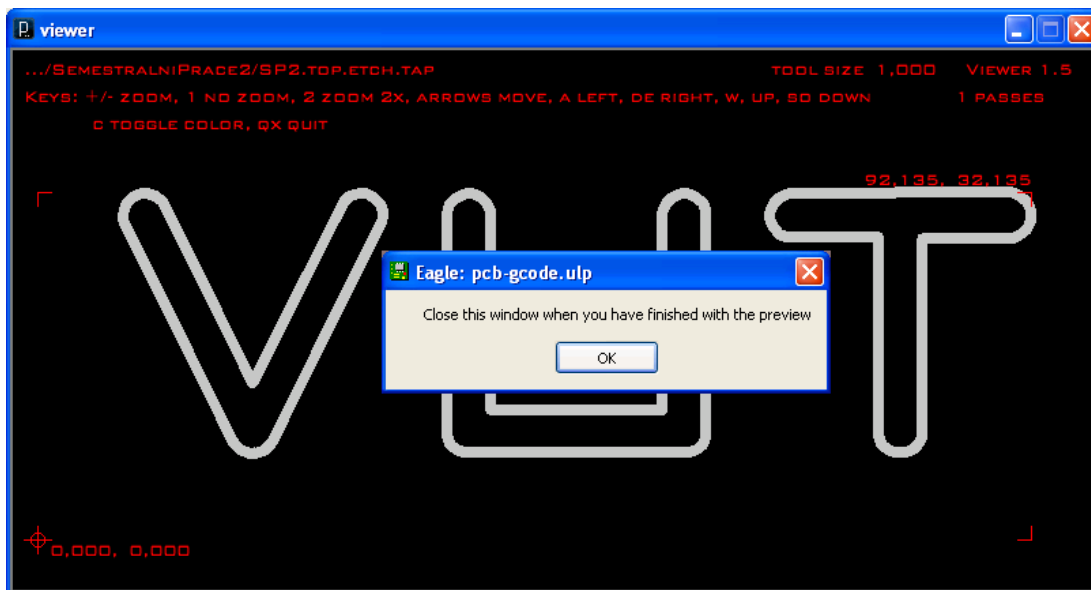
### 5.2.2.6 Other

„Show process menu“ zobrazí jednotlivé kroky generování kódu v Eaglu. Do připraveného okna můžeme zapsat cestu k souboru pro obnovení nastavení souboru menu. Nachází se zde také tlačítko pro reset do továrního nastavení.



Obrázek č. 5.5 – Prostředí PCB GCODE III.

Po nastavení zmíněných hodnot spustíme generování G-kódu tlačítkem „Accept and make my board“. Zobrazí se nám návrh naší desky s oknem pro její potvrzení (obr. č.: 5.6).



Obrázek č. 5.6 – Návrh desky

Po potvrzení začne generování G-kódu. Pokud jste postupovali podle návodu, výsledný soubor bude uložen ve složce s projektem pod názvem „SP2.top.etch.NC“. Nyní si popíšeme jednotlivé části kódu v souboru. Hned na začátku vygenerovaného souboru se nachází komentář s informacemi, které jsme v nastavení programu zaškrtnuli.

```
(.../Program Files/EAGLE-5.6.0/ulp/pcb-gcode.ulp)
(Copyright 2005 - 2009 by John Johnson)
(See readme.txt for licensing terms.)
(This file generated from the board:)
(.../eagle/SemestralniPrace2/SP2.brd)
(Current profile is .../EAGLE-5.6.0/ulp/profiles/emc.pp )
(This file generated 15.12.2011 21:57:53)
```

Následuje definice absolutního programování a nastavení jednotek na milimetry.

```
G21
(Absolute Coordinates)
G90
```

Nyní již můžeme vidět jednotlivé příkazy pro frézování. V pořadí shora rychloposuv na pozici X 0 Y 0, spuštění rotace nástroje, časová prodleva pro ustálení otáček nástroje, nájezd frézy nad povrch, nájezd frézy do počátečního bodu prvního frézování a spuštění frézy do frézovací hloubky zadanou rychlostí.

```
G00 X0.0000 Y0.0000
M03
G04 P5.000000
G00 Z2.0000
G00 X8.0254 Y29.4592
G01 Z-0.2000 F300.00
```

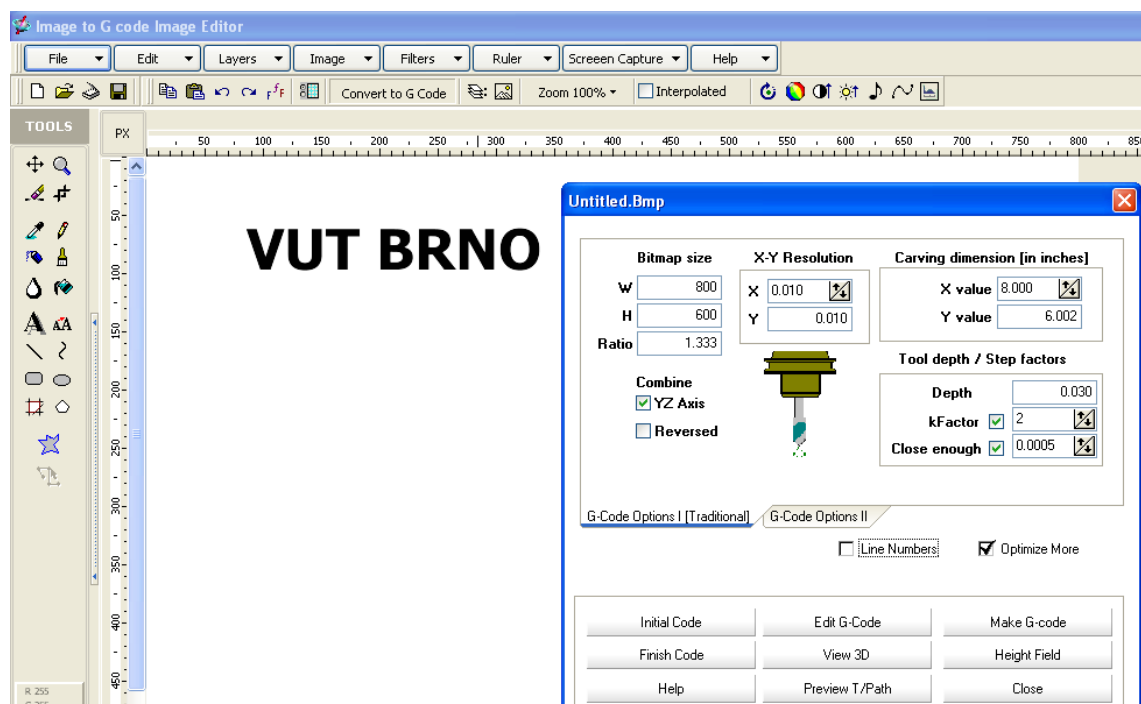
Následné příkazy určují dráhu, po které se bude fréza pohybovat, dokud nenarazíme na příkaz *G00 Z2.0000*, který zvedne frézu nad povrch obrobku. Na konci programu se nachází zakončovací příkazy. Tyto příkazy zvednou frézu do bezpečné výšky nad obráběný povrch *G00 Z10.0000*, zastaví rotaci nástroje *M05* a ukončí program *M02*. Celý soubor přikládám jako elektronickou přílohu spolu s programem na jeho vytvoření a vytvořeným projektem v programu Eagle.

## 5.3 Ostatní programy

Pro úplnost přidávám ještě další programy pro tvorbu G-kódu z bitmapového formátu BMP. Tyto programy ovšem tvoří kompaktní g-kód (neopakuje se druh příkazu např. *G01*, ale jen jeho parametry), a proto by bylo nutné upravit program manipulátoru tak, aby si pamatoval předchozí instrukci.

### 5.3.1 Image to G-code [41]

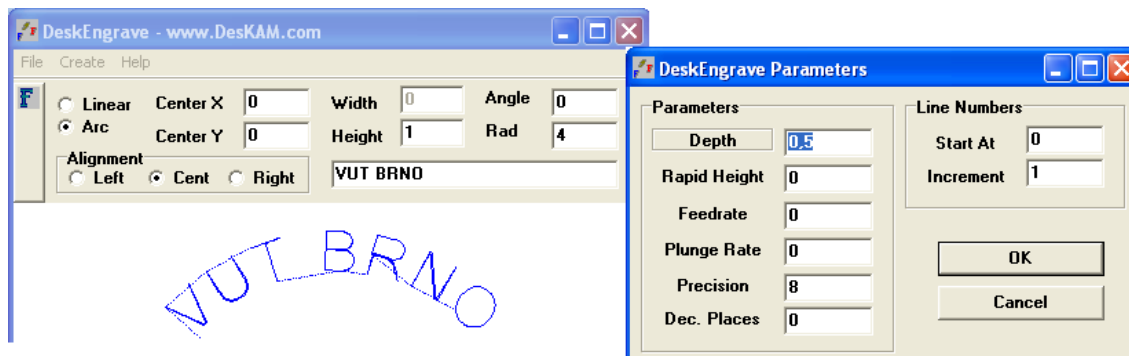
Tento program se nabízí s licencí shareware a po skončení doby 30 dnů je možné jej za 79 dolarů zakoupit. Za tuhle cenu dostaneme velmi přehledné a vybavené prostředí pro nakreslení nebo importování vlastního obrázku. Následně poté můžeme obrázek exportovat do G-kódu, při čemž nám program nabídne celou řadu nastavení, kde můžeme nastavovat podobné parametry jako u předchozího programu.



Obrázek č. 5.7 – Prostředí Image to G-code [41]

### 5.3.2 DeskEngrave [42]

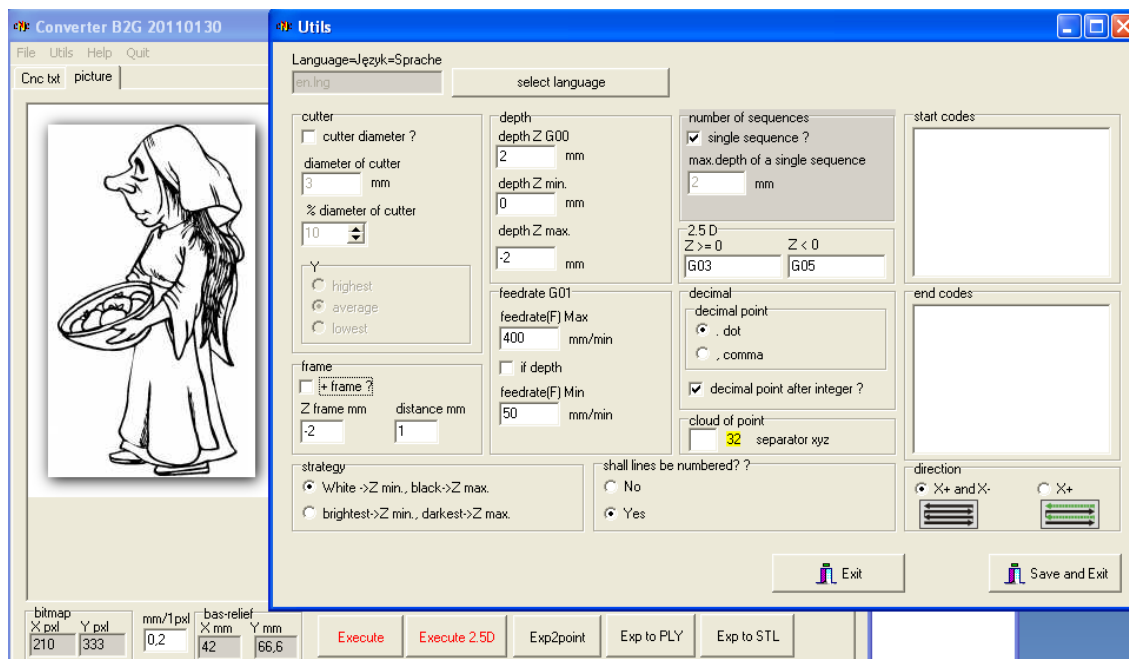
Tento program umožňuje napsat text v řádku nebo do kruhu a poté jej přetvořit na soubor s g-kódem. Jedná se o velice jednoduchý program s malou možností nastavení. Licenci tohoto programu se mi nepodařilo zjistit.



Obrázek č. 5.8 – Prostředí DeskEngrave [42]

### 5.3.3 B2G [43]

Poslední program, který představím, se jmenuje B2G a slouží ke konverzi obrázku ve formátu BMP do g-kódu. Jak lze vidět na následujícím obrázku, po importu vybraného obrázku můžeme generovat kód s nejrůznějšími nastaveními jako třeba hloubka frézování nebo rychlosti posuvů. Program umožňuje generovat kód pro 2,5D i 3D frézky. Licence umožňuje volné použití i pro komerční účely.



Obrázek č. 5.9 – Prostředí B2G [43]

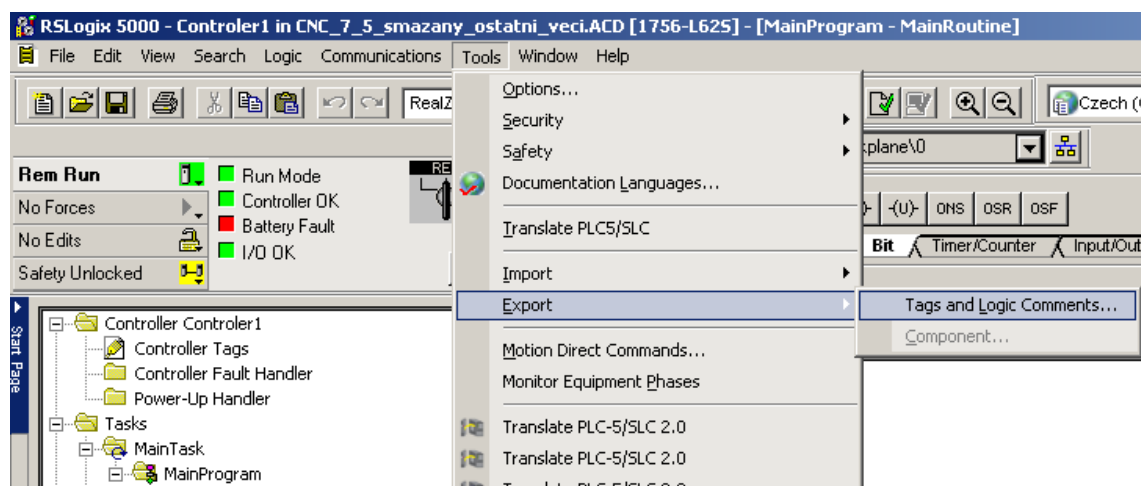
## 6 PŘENOS DAT DO PLC

### 6.1 DDE server [44], [45]

Aplikace DDE (Dynamic Data Exchange) server slouží pro výměnu dat mezi zařízeními v procesu řízení a aplikacemi v systému Windows. V naprosté většině se jedná o výměnu dat mezi PLC a jednotkami (SCADA, HMI) sloužícími pro komunikaci operátora s procesem. Pro komunikaci je nutné mít DDE server, který komunikuje s PLC a následně potřebná data poskytuje DDE klientům. Existuje více variant DDE serveru a to DDE Fast, Advance DDE a NetDDE. DDE server je již poměrně zastaralý, jeho nevýhodou je malá rychlost výměny dat a kompatibilita pouze s platformou Windows. Jeho výhodou je velice jednoduché nastavení a fakt, že DDE klient je integrován do programů, jako je MS Word nebo Excel. Jako novější možnosti komunikace se v dnešní době používá OLE (Object Linking and Embedding) a OPC (Open Process Control). Pro naši aplikaci jsme se rozhodli využít právě DDE server v kombinaci s programem MS Excel, který obsahuje DDE klienta.

### 6.2 Nastavení DDE serveru [46]

Nyní si popíšeme konfiguraci DDE serveru v prostředí RSLinx. Pro nastavení DDE komunikace je nutné nejprve exportovat tagy z prostředí RSLogix 5000 proto, aby DDE server věděl jaké proměnné se v PLC nachází. Export provedeme v prostředí RSLogix 5000, v záložce „Tools“ vybereme položku „Export“ a následně „Tags and logic Comments...“. (obr. č. 6.1)



Obrázek č. 6.1 – Export tagu pro DDE server

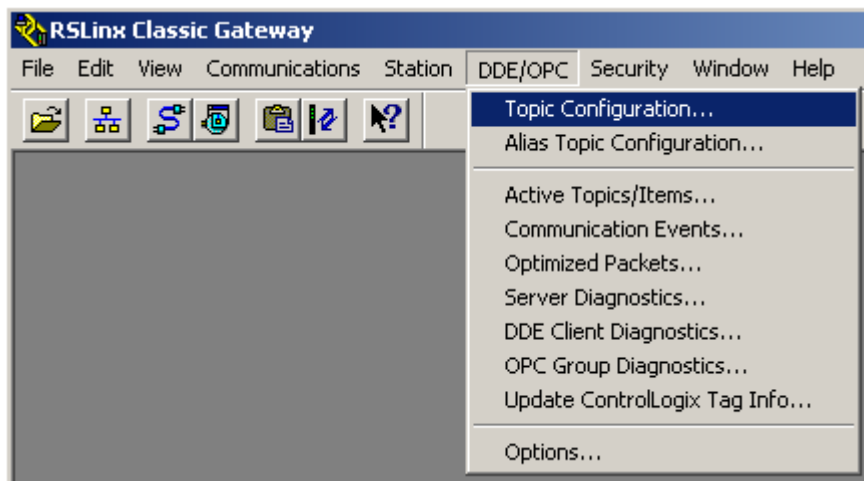


Zobrazí se nám dialogové okno s výběrem místa pro uložení souboru s tagy. Soubor s příponou CSV si pojmenujeme názvem projektu CNC\_7\_5 s příponou Tags. (obr. č. 6.2)



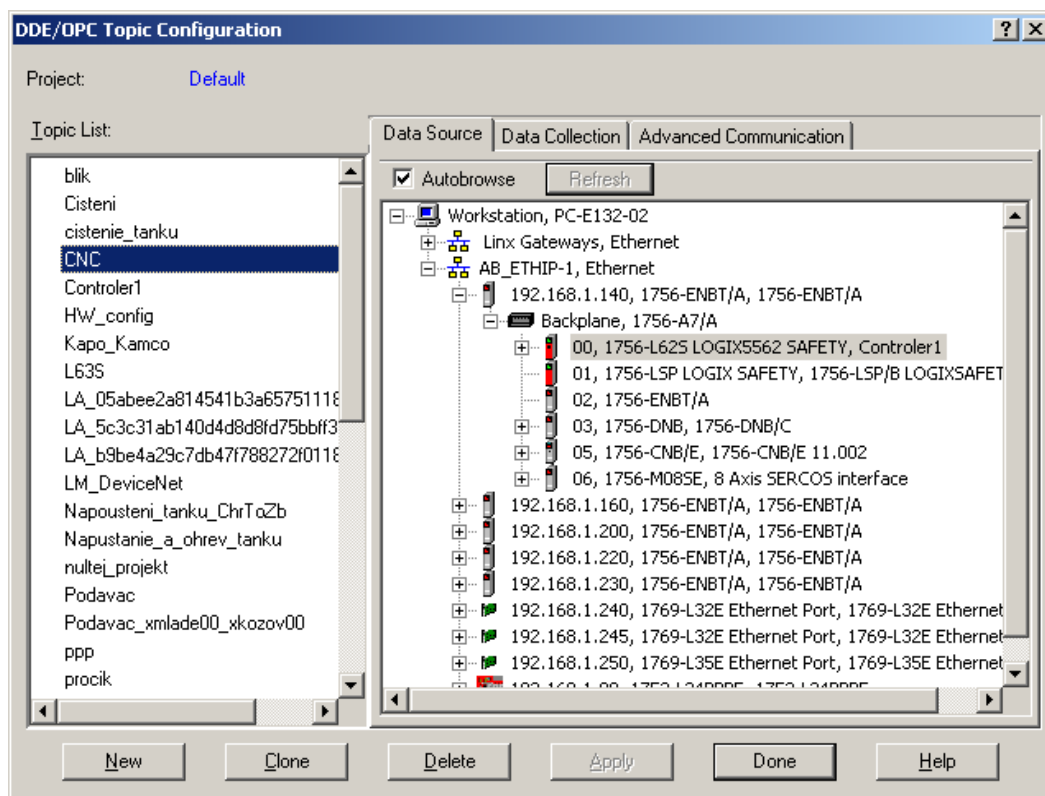
**Obrázek č. 6.2 – Uložení tagů pro DDE server**

Nyní otevřeme prostředí RSLinx a vybereme záložku „DDE/OPC“ a v ní „Topic Configuration..“ (obr. č. 6.3)

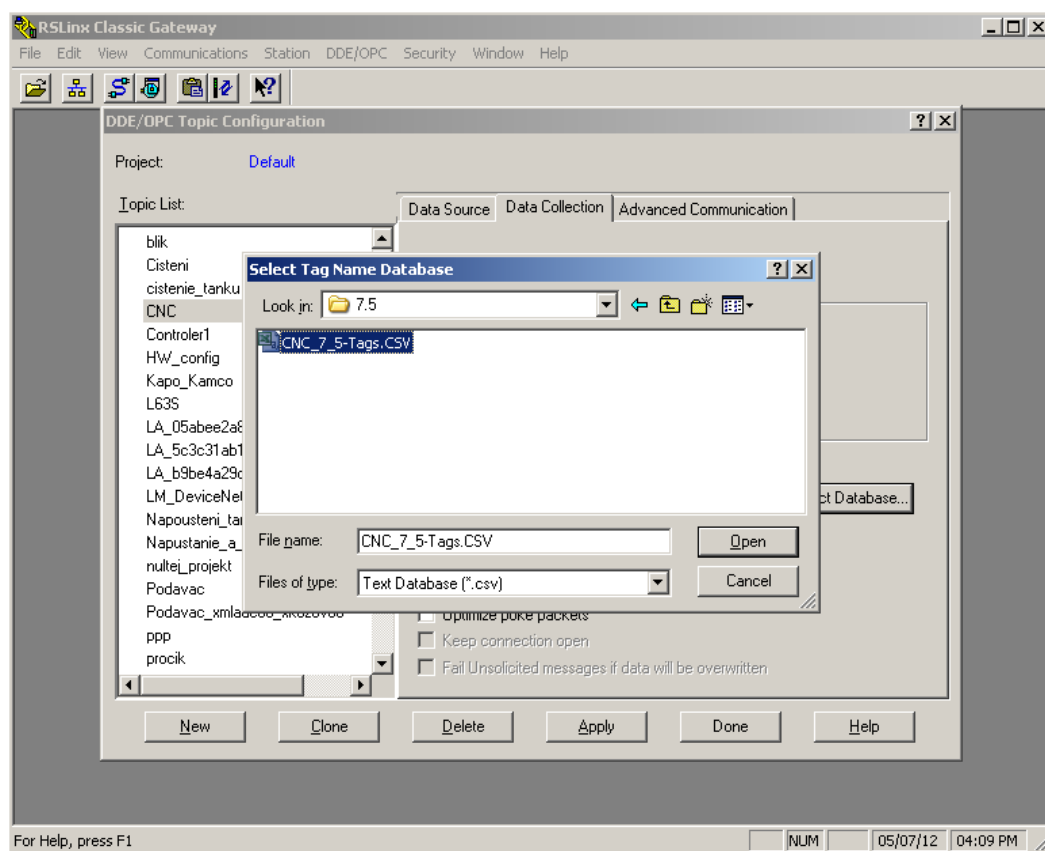


**Obrázek č. 6.3 – Konfigurace DDE v RSLinx**

Otevře se nám okno, kde vytvoříme novou konfiguraci a nastavíme jméno, cestu k procesoru a v další záložce „Data Collection“ nahrajeme náš exportovaný soubor s tagy. Po dokončení tohoto kroku již máme DDE server nastaven a můžeme využívat jeho služeb. (obr. č. 6.4 a 6.5)



Obrázek č. 6.4 – Založení DDE serveru



Obrázek č. 6.5 – Další konfigurace DDE serveru

## 6.3 Program v MS Excel [46]

Poslední část řetězce programového vybavení pro generování a přenos dat do PLC tvoří dokument MS Excel (obr. č. 6.6). Jak již jsem zmiňoval, výhoda DDE serveru spočívá v integraci jeho klientů do prostředí MS Excel. Jedná se tedy o dokument MS Excel s makry pro načtení souboru instrukcí a jeho následné zapsání pomocí DDE serveru do paměti PLC. Náhled tohoto dokumentu vidíme na následujícím obrázku č. 6.6. Nalezneme zde čtveřici tlačítek pro ovládání tohoto programu a to výběr souboru, načtení vybraného souboru do prostředí Excelu, zapsání načtených dat do PLC a smazání načtených dat. Kromě tlačítek se v prostředí nachází i pole zobrazující informace o datech jako umístění zdrojového souboru, jeho velikost, počet načtených instrukcí, čas a počet naposledy zapsaných instrukcí. Ve sloupci „F“ se zobrazují načtené instrukce.

	A	B	C	D	E	F	G
1		<b>Zadejte cestu k souboru!</b>				(.../Program Files/EAGLE-5.6.0/ulp/pcb-gcode.ulp)	
2		<b>Cesta k souboru:</b>	C:\g_kod.txt			(Copyright 2005 - 2009 by John Johnson)	
3		Vybrat cestu k souboru				(See readme.txt for licensing terms.)	
4		<b>Načteno řádků:</b>	124			(This file generated from the board:)	
5						(.../eagle/SemestralniPrace2/SP2.brd)	
6		<b>Velikost souboru [b]:</b>	2944			(Current profile is .../EAGLE-5.6.0/ulp/profiles/emc.pp )	
7						(This file generated 27.12.2011 21:57:53)	
8		<b>Načti soubor</b>				G21	
9						(Absolute Coordinates)	
10		Naposled načteno v:	4:53:37 odp.			G90	
11						G00 X0.0000 Y0.0000	
12		<b>Smazat</b>				M03	
13						G04 P5.000000	
14		<b>Zapiš</b>				G00 Z2.0000	
15						G00 X8.0254 Y29.4592	
16						G01 Z-0.2000 F300.00	
17						G01 X18.1400 Y9.2300 F500.00	
18						G01 X18.2213 Y8.9860	
19		Zapisuji:				G01 X18.3173 Y8.8754	
20		Zapsano:	0			G01 X18.3826 Y8.7447	

Obrázek č. 6.6 – Prostředí pro zápis dat do PLC

### 6.3.1 Výběr souboru s kódem

Pro výběr souboru s instrukcemi, který byl vygenerován pomocí prostředí PCB-GCODE, klikneme na tlačítko s nápisem „Vybrat cestu k souboru“. Zobrazí se dialogové okno, ve kterém si vybereme požadovaný soubor a potvrdíme. Cesta k tomuto souboru se zobrazí v políčku „C2“.

### 6.3.2 Načtení vybraného souboru

Po vybrání souboru můžeme přistoupit k jeho načtení do prostředí Excelu. Po stisknutí tlačítka s názvem „Načti soubor“ se instrukce ze souboru přepíší do sloupce „F“. Zároveň se v políčku „C4“ zobrazí informace o počtu načtených instrukcí, v políčku „C6“ velikost souboru v bitech a čas načtení se zapíše do políčka „C10“.

Při stisknutí tohoto tlačítka se zavolá funkce, která otevře soubor na dříve vybrané adrese a pomocí cyklu DO - WHILE načítá jednotlivé řádky souboru do sloupce „F“. Po dosažení konce souboru zapíše za poslední instrukci zarážku, zjistí a zapíše velikost souboru, počet načtených instrukcí a čas načtení. Během své činnosti inkrementuje proměnnou pomocí, které v případě chyby rozpozná a ohlásí, o jakou chybu se jedná. Soubor s kódem přikládám jako přílohu na CD.

### 6.3.3 Zapsání instrukcí do PLC

Pro zapsání programu do pole v paměti PLC je nutné nejdříve navázat spojení pomocí DDE serveru. Protože spojení během procesu navazujeme vícekrát, vytvoříme jej jako funkci, kterou následně voláme z více míst v kódu. Vytvořená funkce naváže spojení s DDE serverem a ověří, zda je v pořádku.

```
Private Function OpenRSLinx()  
    On Error Resume Next  
    'Otevře spojení s RSLinx  
    OpenRSLinx = DDEInitiate("RSLINX", "CNC")  
    'zjistí, jestli bylo spojení vytvořeno  
    If Err.Number <> 0 Then  
        MsgBox "Error Connecting to topic", vbExclamation, "Error"  
        OpenRSLinx = 0 'Vrati "false" pokud nastala chyba  
    End If  
End Function
```

Pokud již spojení bylo navázáno, dotážeme se na proměnnou v PLC, která koordinuje zápis instrukcí s během programu v PLC. V případě, že je momentálně možné nahrávat instrukce, ověříme, zda počet instrukcí nepřesahuje velikost pole v paměti PLC (max. 17 000). Následně v cyklu zapíšeme hodnoty instrukcí do PLC a jejich počet do připravené proměnné. Nakonec ukončíme spojení s DDE serverem. Pokud nelze zrovna zapisovat do PLC nebo je instrukcí příliš mnoho, informujeme uživatele o vzniklé situaci.

### 6.3.4 Smazání načtených instrukcí

Po stisku tlačítka „Smazat“ vyčistíme prostředí MS Excel. Vymažeme instrukce, jejich počet, velikost souboru, čas posledního načítání i počet zapsaných instrukcí do PLC.

## 7 PROGRAM V PLC

Pro použití modelu manipulátoru jako CNC frézky bylo kromě úprav popsanych v kapitole č. 4 nutné také naprogramovat jeho PLC. Program musí zajišťovat komunikaci s uživatelským rozhraním pro zadávání příkazů obsluhy a nahrávání kódu s instrukcemi. Tyto příkazy a instrukce musí zpracovávat, kontrolovat, a pokud je to možné a bezpečné, také vykonávat. Vykonávání instrukcí mimo jiné vyžaduje i komunikaci s ostatními komponenty manipulátoru, jako jsou servoměniče nebo I/O periferie. Program je napsán v programovacím jazyku Ladder diagram, který vzdáleně připomíná reléovou logiku, protože byl původně navrhnout s ohledem na srozumitelnost pro tehdejší programátory pracující s reléovými systémy. V této kapitole si podrobněji popíšeme vytvořený program a jeho možnosti. Program nalezneme na přiloženém CD ve formě souboru pro nahrání do PLC i ve formátu PDF.

### 7.1 Konfigurace PLC a komunikace

Základní konfigurace řídicího automatu a nastavení servoměničů byla převzata z bakalářské práce ŘÍZENÍ 3 OS MANIPULÁTORU od pana Bc. Miroslava Matějčička [13]. V převzaté konfiguraci jsem již našel nakonfigurované PLC (viz tabulka č. 7.1) spolu s rozhraním Ethernet, DeviceNet a Sercos. Dále zde byly pomocí laboratorní sítě Ethernet/IP nakonfigurovány vzdálené vstupy náležící jinému PLC, připojenému k laboratornímu simulátoru. Nejdůležitějším prvkem v tomto konfiguračním nastavení bylo nastavení jednotlivých servoměničů a jejich následné svázání do souřadného systému.

Pozice:	Katalogové č.:	Popis:
0	1756-L62S	Procesor
1	1756-LSP	Safety procesor
2	1756-ENBT/A	Ethernet/IP interface
3	1756-DNB	DeviceNet interface
6	1756-M08SE	Sercos interface

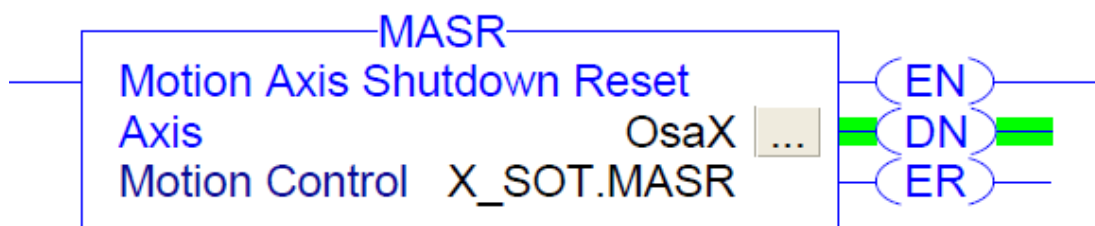
Tabulka č. 7.1 – Konfigurace PLC

### 7.2 Instrukce pro servoměniče [30]

Snad nejdůležitější částí vytvořeného programu bude řízení pohybu os manipulátoru. Servoměniče Kinetix 2000 použité v našem případě lze použít pro koordinovaný pohyb ve více osách. Výrobce proto nabízí instrukce pro jednotlivé osy zvlášť, ale také pro celý souřadný systém. Nyní si představíme základní instrukce pro řízení os manipulátoru použité v našem programu.

### 7.2.1 MARS

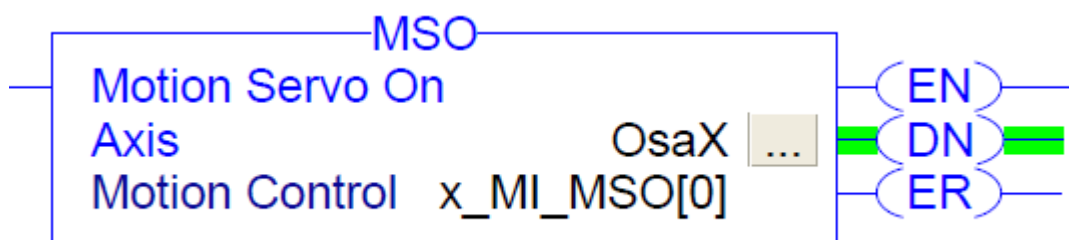
MARS (Motion Axis Shutdown Reset) - tato instrukce (obr. č. 7.1) smaže všechny hlášené poruchy na zvolené ose, a pokud se osa nacházela ve vypnutém stavu, připraví ji na zapnutí. Instrukce požaduje dva parametry. První parametr „Axis“ určuje osu, které se tato instrukce týká a druhý je „Motion Control“, ten určuje strukturu typu „Motion instruction“ určenou pro zápis výsledků této instrukce. Instrukce MARS signalizuje na svém výstupu tři stavy. První stav označený ve struktuře EN signalizuje logickou jedničku na vstupu instrukce. Druhý označený DN signalizuje provedení dané instrukce bez chybového hlášení a třetí ER signalizuje chybový stav instrukce.



Obrázek č. 7.1 – MASR [30]

### 7.2.2 MSO

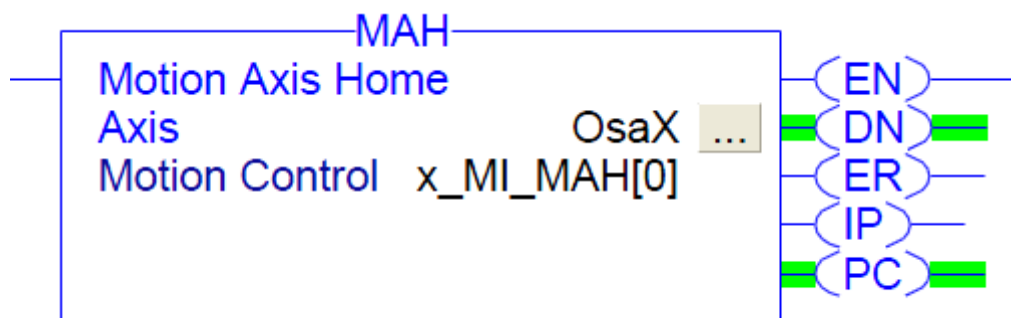
Instrukce MSO (Motion Servo On) aktivuje osu a připraví ji k používání. Vstupní i výstupní parametry se shodují s instrukcí MASR. Aby aktivace osy proběhla v pořádku, nesmí se na ose vyskytovat chybové hlášení, proto je dobré a v některých případech nezbytné před instrukcí MSO použít instrukci MARS.



Obrázek č. 7.2 – MSO [30]

### 7.2.3 MAH

Instrukce MAH (Motion Axis Home) zaparkuje zvolenou osu do výchozí polohy, na které se nachází snímač. Zároveň se také provede inicializace souřadného systému osy. Pokud by po zapnutí serva tato instrukce nebyla použita, nedošlo by ke kalibraci serva, nejspíše dojde k havárii způsobené chybným hlášením o poloze dané osy. Instrukce MAH (obr. č. 7.3) signalizuje již zmíněné stavy a přidává stavy IP signalizující právě probíhající instrukci a PC značící dokončení instrukce.



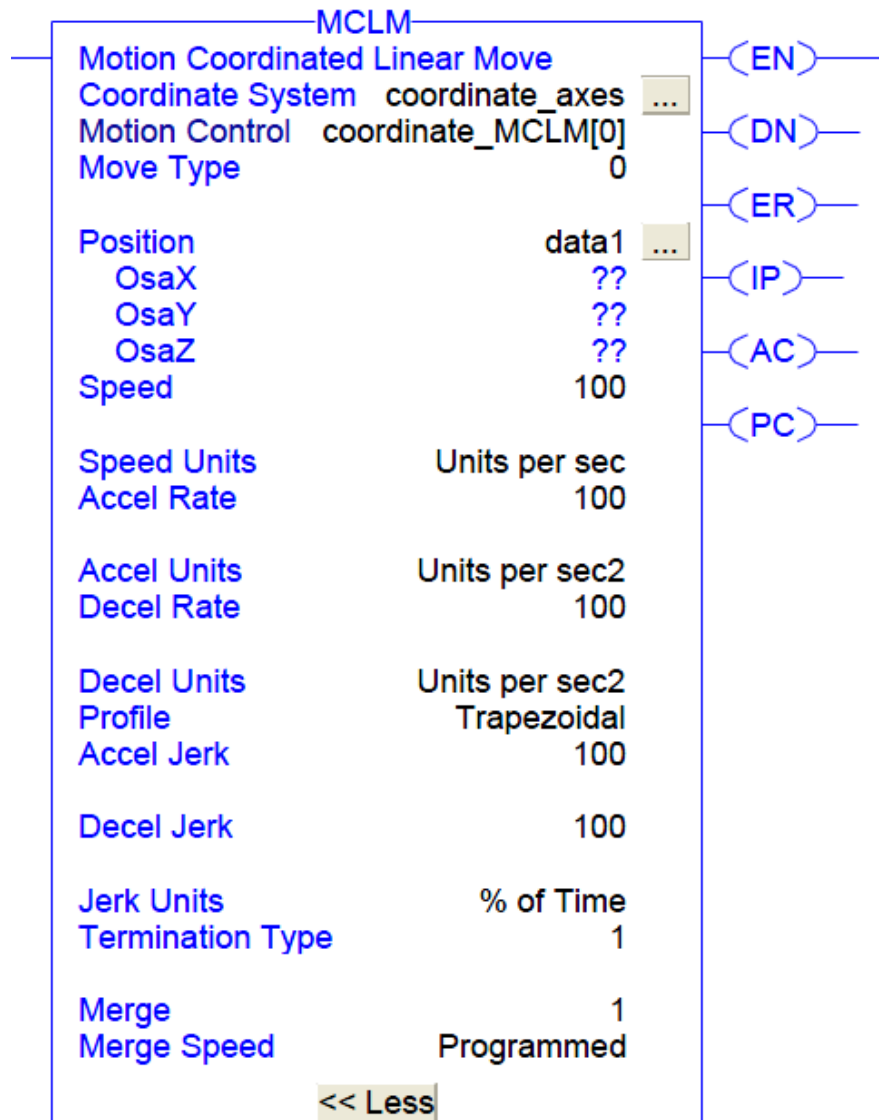
Obrázek č. 7.3 – MAH [30]

## 7.2.4 MCLM

Nyní se konečně dostáváme k instrukci (obr. č. 7.4) provádějící požadovaný pohyb našeho souřadného systému. Jedná se o lineární pohyb z výchozího místa na místo určené souřadnicemi této instrukci. První parametr potřebný ke správnému vykonání tohoto pohybu je určení souřadného systému, který má tento pohyb vykonat, následuje opět struktura pro zapsání stavů instrukce. Parametr „Move Type“ určuje, zda se jedná o přírůstkové (1) nebo absolutní (0) programování. Do parametru „Position“ uvedeme strukturu, ve které se nachází souřadnice koncového bodu. Následující proměnné této instrukce se týkají rychlosti posunu „Speed“ a „Speed Units“, zrychlení „Accel“ a „Accel Units“, zpomalení „Decel“ a „Decel Units“. Proměnná „Profile“ rozhoduje o použití S-křivky nebo trapézového zrychlování (zpomalování). „Accel Jerk“, „Decel Jerk“ definují tvar S-křivky, pro trapézový profil nemají vliv. V případě použití více instrukcí pohybu za sebou rozhoduje parametr „Termination Type“ o jejich navázání viz tabulka č. 7.2.

Hodnota:	Název:	Popis ukončení instrukce:
0	Actual tolerance	při dosažení cílové pozice a setrvání v toleranci
1	No Settle	při dosažení cílové pozice
2	Command Tolerance	při dosažení pozice (cílová pozice - Command Tolerance)
3	No Decel	při dosažení zpomalovacího bodu
4	Follow Contour Velocity Constrained	při dosažení cílové pozice
5	Follow Contour Velocity Unconstrained	při dosažení cílové pozice

Tabulka č. 7.2 – Parametr Termination Type [30]



Obrázek č. 7.4 – MCLM [30]

### 7.2.5 MCCM

Instrukce MCCM (obr. č. 7.6) se velmi podobá předchozí instrukci jen s rozdílem, že místo lineární interpolace provádí kruhovou interpolaci. Většina parametrů se shoduje s instrukcí MCLM, a proto zde uvedu jen parametry rozdílné. Výsledný kruh je dán výchozím bodem, kde se rameno nachází a bodem zadaným do instrukce. Následující parametry ovlivňují, jak bude výsledný pohyb vypadat. První parametr se nazývá „Circle type“ a může nabývat hodnot „0 = Via“, „1 = Centre“, „2 = Radius“ a „3 = Center Incremental“. Pokud zadáme hodnotu „0 = Via“ v poli „Via/Centre/Radius“, očekává procesor hodnotu bodu, přes který bude vykreslená část kružnice procházet. Při hodnotě „1 = Centre“ bod zadaný v poli „Via/Centre/Radius“

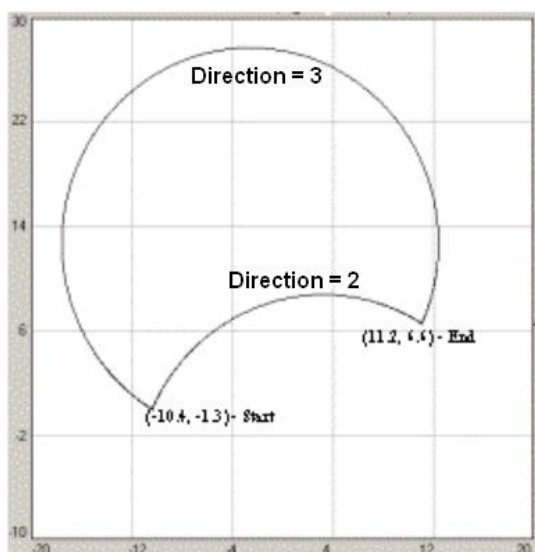


určuje střed této kružnice. Požadujeme-li vykreslení kružnice o zadaném poloměru, zadáme jeho hodnotu do pole „Via/Centre/Radius“ a do parametru „Circle type“ zadáme hodnotu „2 = Radius“. Pokud chceme zadat střed kružnice v přírůstkovém tvaru, nastavíme parametr „Circle type“ na hodnotu „3 = Center Incremental“. Význam parametrů „Circle type“ a „Via/Centre/Radius“ zobrazuje také následující tabulka.

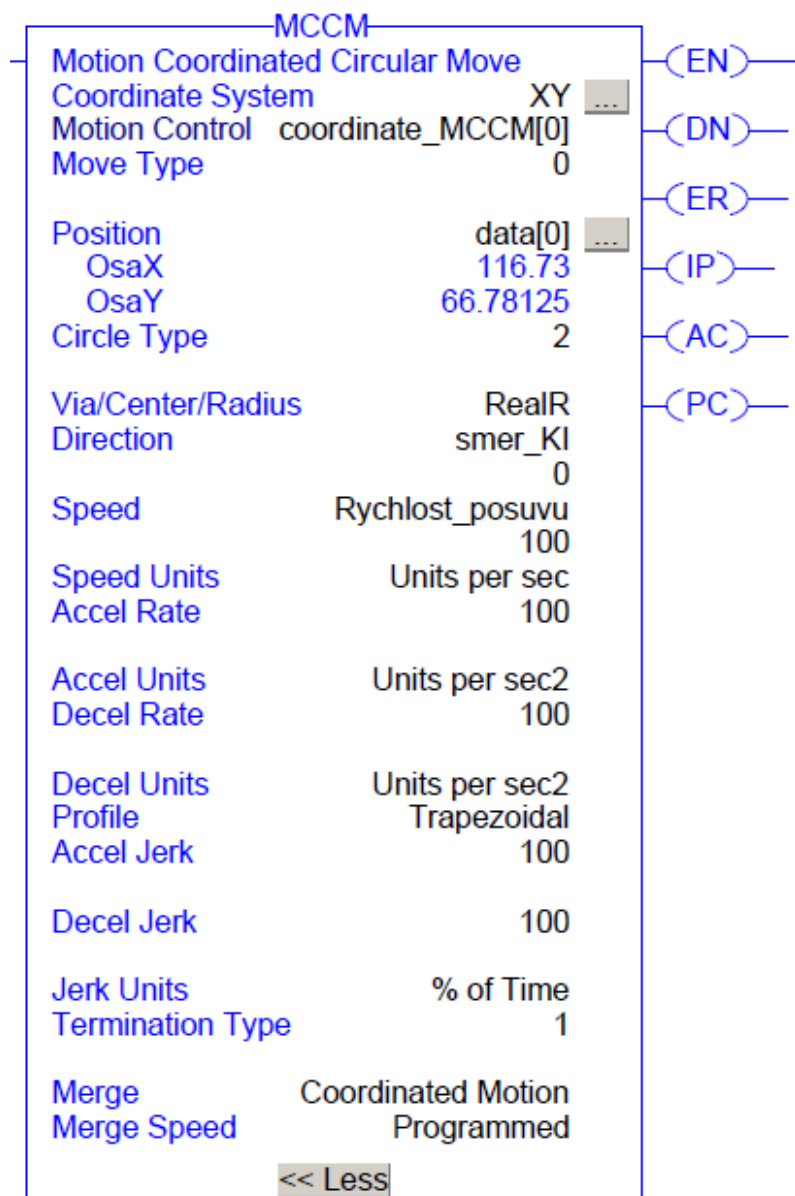
Hodnota:	Název:	Význam parametru „Via/Centre/radius“:
0	Via	průchozí bod
1	Centre	středový bod
2	Radius	poloměr
3	Center Incremental	středový bod v přírůstkovém tvaru

**Tabulka č. 7.3 – Parametr Circle type [30]**

Další odlišný parametr s názvem „Direction“ určuje směr vykreslované kružnice. Hodnota 0 značí vykreslení po směru hodinových ručiček, hodnota 1 směr opačný. Hodnoty 2 a 3 pro zadaný poloměr vysvětluje následující obrázek.



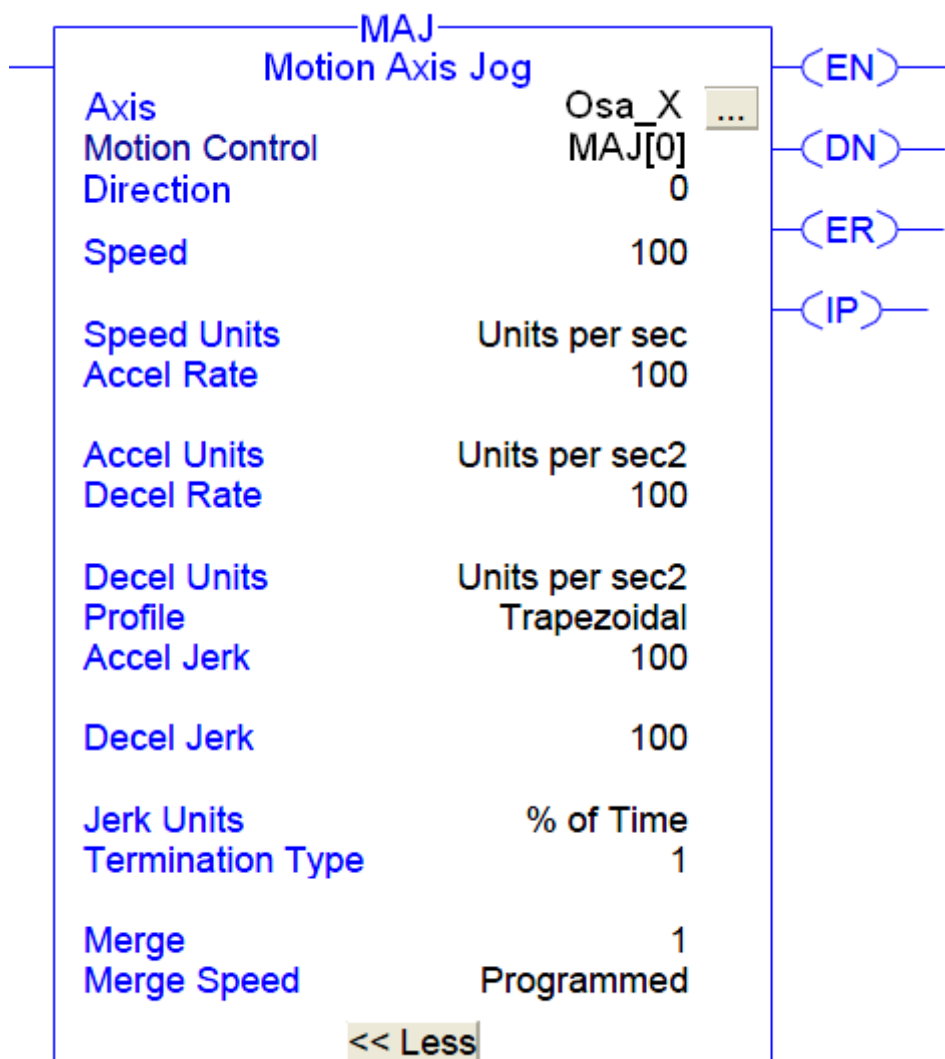
**Obrázek č. 7.5 – Parametr Direction [30]**



Obrázek č. 7.6 – MCCM [30]

## 7.2.6 MAJ

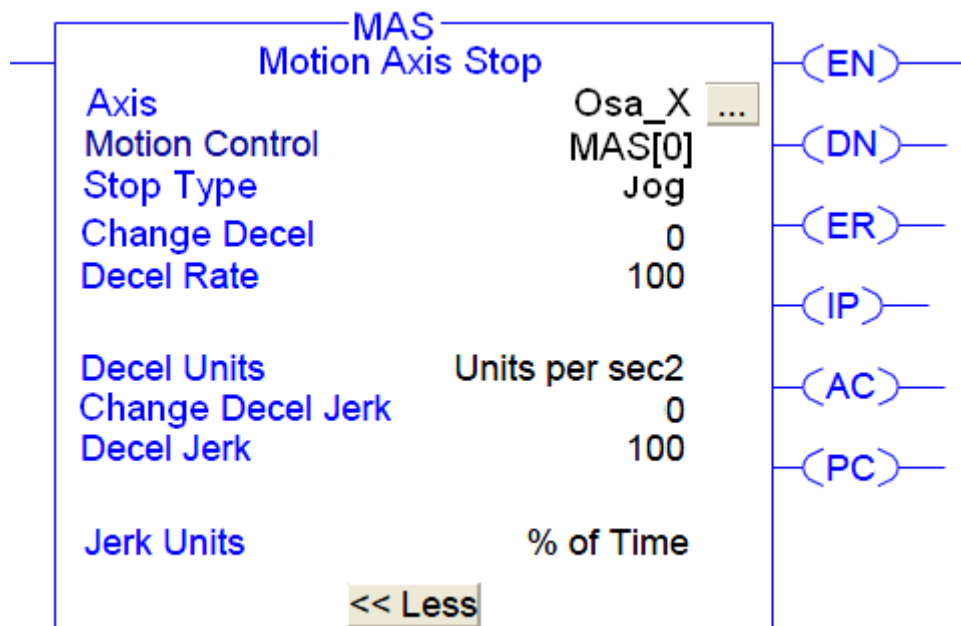
MAJ (Motion Servo Jog) se používá pro ruční posuv osy ve zvoleném směru. Jelikož nemůžeme v ručním posuvu zabezpečit koordinovaný pohyb více os, vztahuje se tato instrukce pouze k jedné ose. Její parametry se ve většině shodují s předchozí instrukcí. Nevybíráme zde ovšem koordinační systém os, ale pouze jednu osu (parametr Axis) a místo cílových souřadnic se zde určuje kladný nebo záporný směr pohybu, (volba Direction) pro hodnotu 0 se pohybujeme vpřed, pro hodnotu 1 vzad. Instrukce MAJ (obr. č. 7.7) spustí posuv se zvolenými parametry, ale po skončení její platnosti se osa nezastaví, jak by se mohlo očekávat. Pro zastavení pohybu osy je třeba použít instrukci MAS (Motion Axis Stop).



Obrázek č. 7.7 – MAJ [30]

## 7.2.7 MAS

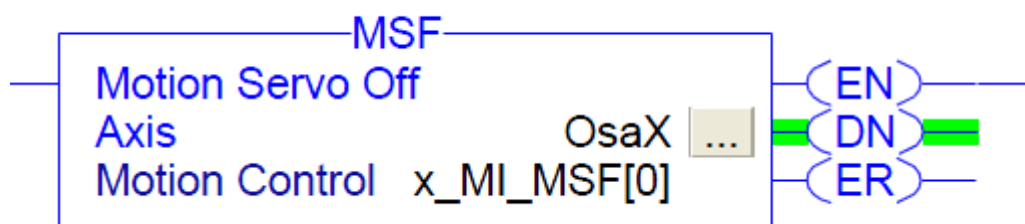
Jak již bylo zmíněno v předchozím odstavci, zastavit osu rozjetou instrukcí MAJ lze pomocí instrukce MAS (Motion Axis Stop). Této instrukci (obr. č. 7.8) je třeba nastavit osu, kterou chceme zastavit a typ instrukce, kterou byla osa spuštěna. Dále si volíme, jestli má k zastavení dojít maximální rychlostí (Change Decel nastavit na hodnotu 0) nebo podle zvolených parametrů, jejichž význam jsme si již zmiňovali.



Obrázek č. 7.8 – MAS [30]

### 7.2.8 MSF

Instrukce MSF (Motion Servo Off) uvede osu do vypnutého stavu. Osa setrvává v tomto stavu, dokud není spuštěna instrukce MSO (Motion Servo On), která osu opět uvede do aktivního stavu. Vstupní parametry jsou osa a struktura pro uložení stavu instrukce, výstupní parametry EN, DN, ER jsem již vysvětloval.



Obrázek č. 7.9 – MSF [30]

## 7.3 Popis programu

Program se dělí do několika částí. Tyto části jsou podle potřeby volány z hlavní smyčky a zajišťují jednotlivé funkční celky. Příkladem může být podprogram pro ruční ovládání nebo pro automatický režim. V následujících odstavcích si popíšeme, jak pracuje hlavní smyčka i jednotlivé podprogramy.

### 7.3.1 Hlavní smyčka

Po spuštění PLC se automaticky zavolá hlavní smyčka, ta po svém spuštění jednou zavolá podprogram pro inicializaci proměnných a instrukci MASR (Motion Axis Shutdown Reset), která restartuje osu a smaže všechna její chybová hlášení. Instrukce MASR se vztahuje pouze k jedné ose, a proto se instrukce volá pro každou osu zvlášť.

Po inicializační sekvenci se zde nachází volání podprogramu s názvem „tlacitka“, který se stará o zjištění stavů vstupů s tlačítky na laboratorním simulátoru a jejich zápisu do příslušných proměnných. Tato rutina byla používána pro ovládání manipulátoru před vytvořením uživatelského rozhraní a nyní je její volání zablokováno.

Pokud by se instrukce MASR skončily s chybovým hlášením, nabízí se zde uživateli také možnost manuálního resetu os. Pokud předchozí instrukce proběhly bez chybových hlášení, následuje trojice instrukcí MSO (Motion Servo On), po jejímž provedení se jednotlivé osy aktivují a jsou připraveny k použití. Jako první operace s osami je nutné je kalibrovat ve výchozí pozici. To se provádí instrukcemi MAH (Motion Axis Home). Pokud uživatel nezvolí kalibraci os, není mu dovoleno s nimi jakkoliv manipulovat. V případě úspěšné kalibrace a absenci chybových stavů os ohlásí proměnná „připraveno“ možnost manipulace s osami.

Následuje trojice podprogramů: automatický režim, ruční posuv a kontrola G-kódu. Automatický režim předpokládá zkontrolování programu a ukončení všech ostatních režimů. Ruční posuv předpokládá pouze ukončený automatický režim a kontrola kódu ještě neprobíhající reset automatického režimu.

Restartování automatického režimu provedeme při vypnutém automatickém režimu na žádost uživatele. Z důvodů své jednoduchosti nebylo nutné ji vytvářet jako podprogram. Jedná se pouze o restartování stavového automatu automatického režimu, vynulování pořadí instrukce a časovače pro zpoždění při roztáčení vřetena. Dále také zastavení vřetena a vynulování proměnné ohlašující dosažení poslední instrukce.

Následující část kódu umožní uživateli odemknout paměť PLC pro nakopírování instrukcí z programu MS Excel. Zároveň se smaže platnost kontroly instrukcí v paměti PLC.

Žádost uživatele o okamžité vypnutí všech os a zastavení vřetene z bezpečnostních důvodů nepodléhá žádné podmínce a je neprodleně vykonána.

Při aktivaci optické závory se osy pouze zastavují. To ovšem neplatí pro ruční posuv, kdy má uživatel přístup do prostoru obrábění. Pro zvýšení bezpečnosti je ovšem rychlost všech os omezena.

Pro účely vizualizace se neustále obnovují proměnné pro aktuální a nahlíženou instrukci. Poslední dva podprogramy zajišťují restartování alarmů a řešení chybových stavů.

### 7.3.2 Ruční režim

Ruční režim je možné spustit pouze, pokud se všechny osy nachází ve stavu připraveném k použití, tzn. po kalibraci a bez chyb, a dále za předpokladu neprobíhajícího automatického režimu. Tento režim řeší především posuv os podle pokynů uživatele. Posuv se provádí instrukcí MAJ (Motion Axis Jog). Tato instrukce je zde pro každou osu dvakrát, a to pro posuv vpřed a vzad. Pokyn uživatele je blokován, pokud se osa nachází v krajní pozici, anebo pokud je momentálně vydán příkaz k jejímu zastavení. V případě jakékoliv chyby nebo překročení limitu se osa okamžitě zastavuje instrukcí MAS (Motion Axis Stop). K zastavení dojde také při protichůdných požadavcích na posuv osy. Při selhání instrukce MAS se program pokusí ještě osy vypnout instrukcí MAF (Motion Axis Off).

V tomto režimu se také určuje nulový bod obrobku, a to zapsáním offsetů jednotlivých os do vyhrazených proměnných.

### 7.3.3 Dekódování instrukce

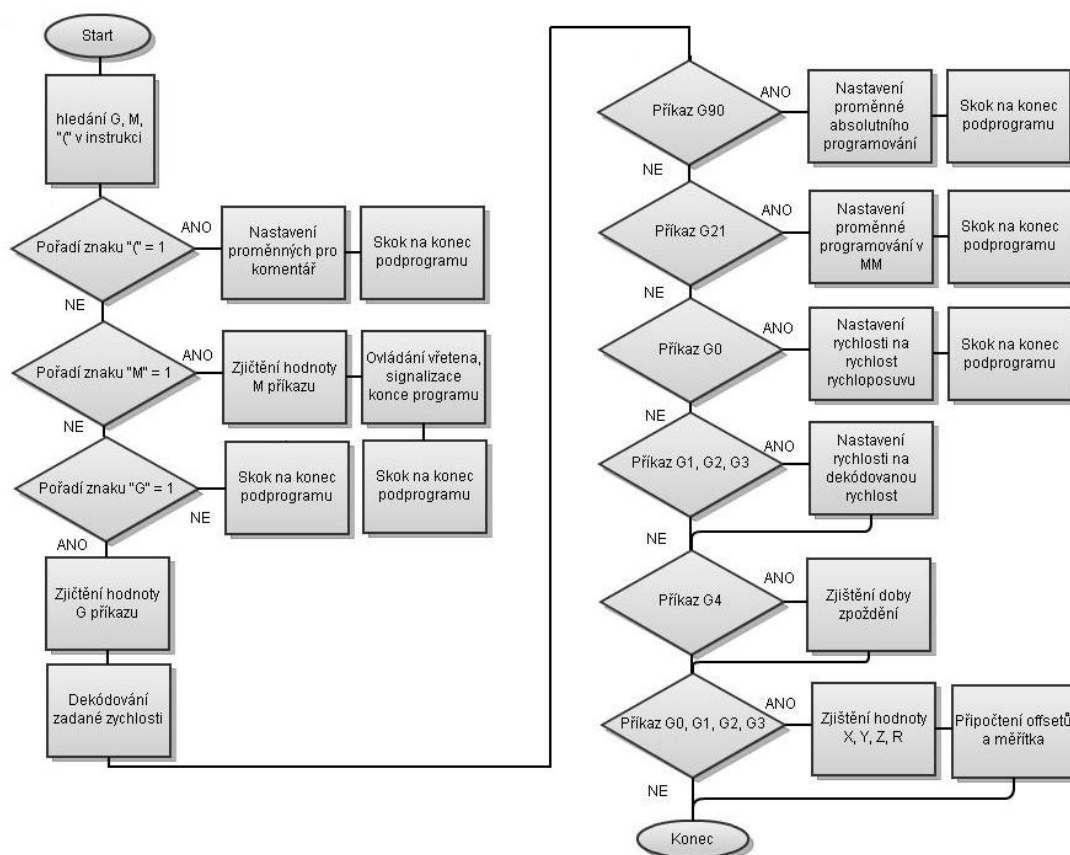
Podprogram pro dekodování instrukce se nespouští přímo z hlavní smyčky, ale volá si jej automatický režim a kontrola instrukcí. Tento program nevyužívá servoměniče, a tak neomezuje volání jiných programů, pouze je nutné zabezpečit, aby nebyl volán vícenásobně, protože při každém volání používá stejnou sadu proměnných. Program slouží k dekodování instrukce, která je zapsána v proměnné typu „STRING“, na kterou ukazuje ukazatel aktuální instrukce. Pod slovem dekodování se rozumí zjištění druhu příkazu, nastavení příznaků pro vykonání instrukce a zapsání hodnot parametrů instrukce do příslušných proměnných.

Po svém spuštění podprogram zjistí, o jaký typ instrukce se jedná. A to vyhledáním znaků „G“, „M“ a „(“ v instrukci. Pokud zjistí výskyt znaku „(“ prohlásí instrukci za komentář, ohlásí dekodování instrukce a odskočí na konec podprogramu. V případě nalezení znaku „M“ se pokusí zjistit hodnotu M instrukce a odskočí na konec podprogramu, kde se tato instrukce řeší. Zde ohlásí rozpoznání M instrukce a podle její hodnoty ovládá rotaci vřetene a ohlašuje instrukci značící konec programu.

Pokud se na prvním místě instrukce nachází písmeno G, opět zjistíme hodnotu následující za tímto písmenem. Pokud se jedná o instrukce, kde je možno volit rychlost posuvu os (G01, G02 nebo G03), hledáme pozici písmene F a následující hodnotu

zapišeme do proměnné určující rychlost pracovního posuvu. Při výskytu instrukce G90 nastavíme absolutní programování a výskyt instrukce G21 značí programování v milimetrech. V případě rychloposuvu je rychlost určena na hodnotu 100 mm/min, pro případ pracovního posuvu bude rychlost určena nově dekodovanou rychlostí nebo poslední dekodovanou rychlostí. Instrukce G04 obsahuje parametr P nesoucí hodnotu zpoždění potřebnou k ustálení otáček vřetene. Tato doba se zadává v sekundách, a protože časovač pracuje s časovým krokem 1ms, je nutné tuto hodnotu násobit tisícem.

Další skupina příkazů zjistí, zda a kde se v instrukci nachází znaky X, Y, Z a R. Pokud naleznou znak X, vykopírují následnou hodnotu, tato hodnota určuje novou pozici pro osu X. Hodnota se násobí měřítkem a zvětšuje o přednastavený offset. Poté proběhne ještě kontrola mezi hodnoty. V případě překročení meze je hodnota nahrazena hodnotou meze, kterou překročila. To vše se opakuje pro osu Y a Z. Nulová hodnota osy Z manipulátoru se nachází zcela nahoře, to ovšem neodpovídá orientaci osy v CNC zařízeních. Proto při určování hodnoty pro tuto osu musíme hodnotu odečítat od jejího maxima. Při použití kruhové interpolace se poloměr určuje parametrem R, jeho hodnota se určuje podobně jako hodnoty pro polohu os. Poslední příkaz rozhoduje, zda byla instrukce dekodována správně, pokud ano, nastaví hodnotu stavového automatu na pokračování. Pokud ne, nastaví chybový stav. Vývojový diagram programu zobrazuje obrázek č.: 7.10.



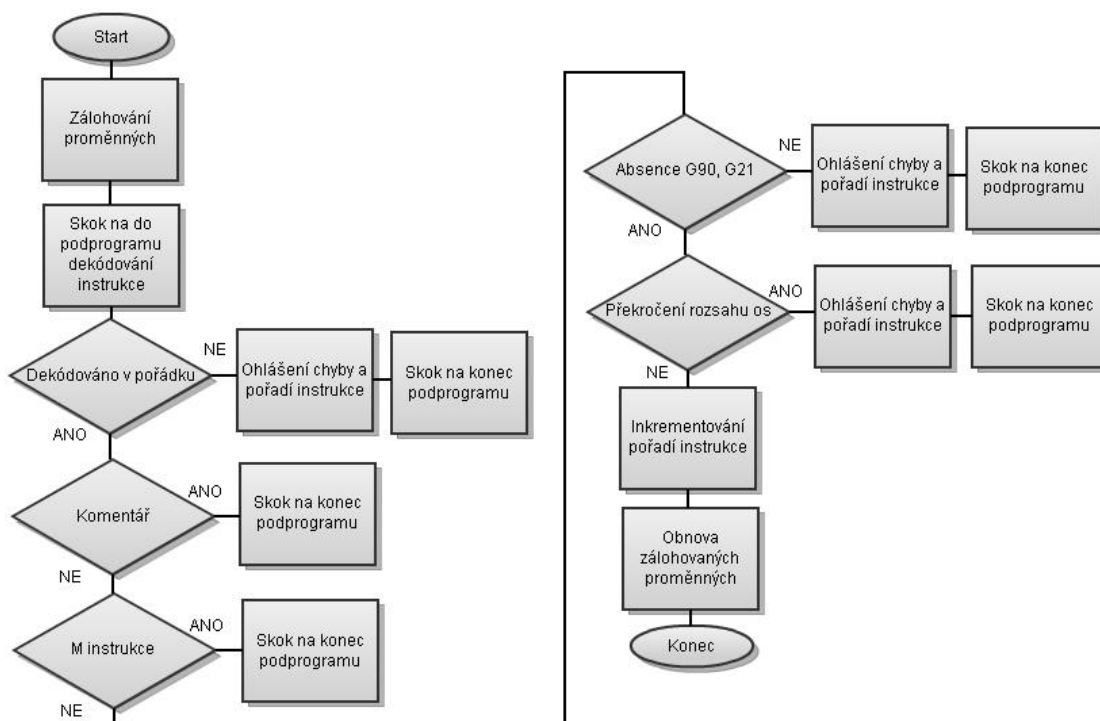
Obrázek č. 7.10 – Dekódování instrukce

### 7.3.4 Kontrola instrukcí

Před spuštěním automatického režimu se vyžaduje kontrola instrukcí. Tato kontrola se vyžaduje pokaždé, když bylo možno do pole s instrukcemi zapisovat. Tato rutina se zaměřuje spíše na zápis instrukcí než na detailní rozbor sledu instrukcí. Neslouží pro zamezení poškození manipulátoru, ale pro eliminaci chybně zapsané instrukce uprostřed obrábění. V tomto případě by zbytečně došlo k vytvoření zmetku.

Prvním krokem této rutiny je zálohování používaných proměnných jako například pořadí instrukce, hodnoty příkazů jednotlivých os anebo rychlost posuvu. Tato záloha se vytváří pro případ, že by bylo nutné kontrolu provádět bez ztráty aktuálního stavu manipulátoru.

Kontrola postupně prochází všechny instrukce a hledá, jestli není některá z nich neplatná. O platnosti instrukce rozhoduje podprogram pro dekódování instrukcí popsany v předchozím odstavci. Po úspěšném dekódování instrukce se dotazujeme na její druh. Komentáře a M instrukce dále nekontrolujeme. G instrukce podrobujeme dalšímu zkoumání. Nejprve se ujistíme, že před první instrukcí posuvu předcházely instrukce G90 (absolutní programování) a G21 (programování v milimetrech) a následně se zjistí, zda parametry instrukce nepřesahují meze manipulátoru. V případě chyby zapíšeme pořadí chybné instrukce a ohlásíme chybu programu. Pokud instrukce splňuje všechny podmínky, pokročíme k další. Na závěr se obnoví dříve zálohované proměnné. Vývojový diagram programu zobrazuje obrázek č.: 7.11.



Obrázek č. 7.11 – Kontrola instrukcí

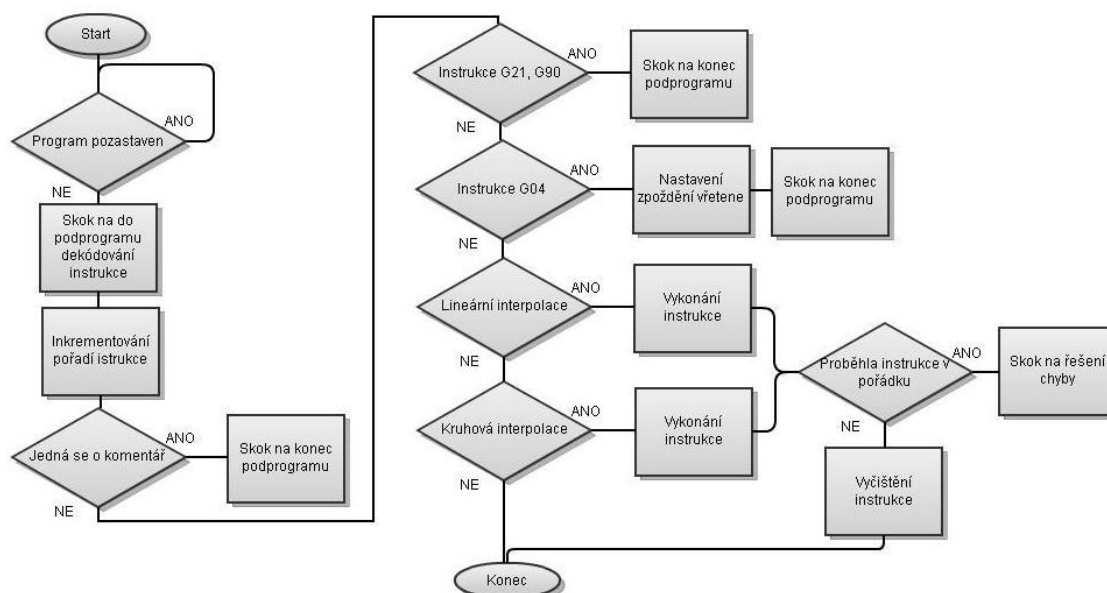


### 7.3.5 Automatický režim

Automatický režim slouží pro vykonávání instrukcí, které se nachází v poli v paměti PLC. Generování a přesun instrukcí do PLC bylo vysvětleno v některé z předchozích kapitol. Před začátkem tohoto programu musí instrukce projít kontrolou a následně nesmí být zpřístupněna možnost jejich změny. Přerušení procesu vykonávání instrukcí zde nalezneme dvakrát. Zaprvé se jedná o přerušení po dokončení instrukce a zadruhé okamžité přerušení, které nastane při narušení obráběcího prostoru.

Jako první testuje program podmínku zastavující vykonávání instrukcí, v případě její splnění nepokračuje dále. Automatický režim ke své práci využívá podprogram pro dekódování instrukce, který se volá za splnění předchozí podmínky. V případě úspěšného dekódování instrukce se posune ukazatel na další instrukci. M instrukce se vykonává již při jejím dekódování, a proto se tyto instrukce spolu s komentáři přeskakují. Výjimkou mezi G instrukcemi tvoří G90 a G21, které se rovněž přeskakují. Instrukce G04 aktivuje časovač určující zpoždění pro ustálení otáček vřetene. Po uplynutí nastavené doby se pokračuje další instrukcí.

Instrukce lineárního pohybu G00 a G01 provádí příkaz MCLM (Motion Coordinated Linear Move), jeho parametry byly nahrány do struktur a proměnných při dekódování instrukce. Po dokončení aktuálního pohybu se instrukce restartuje a připraví pro další použití. Instrukce G02 a G03 se provádí totožně, ovšem příkazem MCCM (Motion Coordinated Circular Move). Dojde-li k chybě při vykonávání instrukcí pohybu, vygeneruje se chybové hlášení, na jehož základě se aktivuje alarm. Po dokončení poslední instrukce se program zastaví a vyčkává na své opětovné spuštění nebo nahrání jiného kódu. Vývojový diagram programu zobrazuje obrázek č.: 7.12.

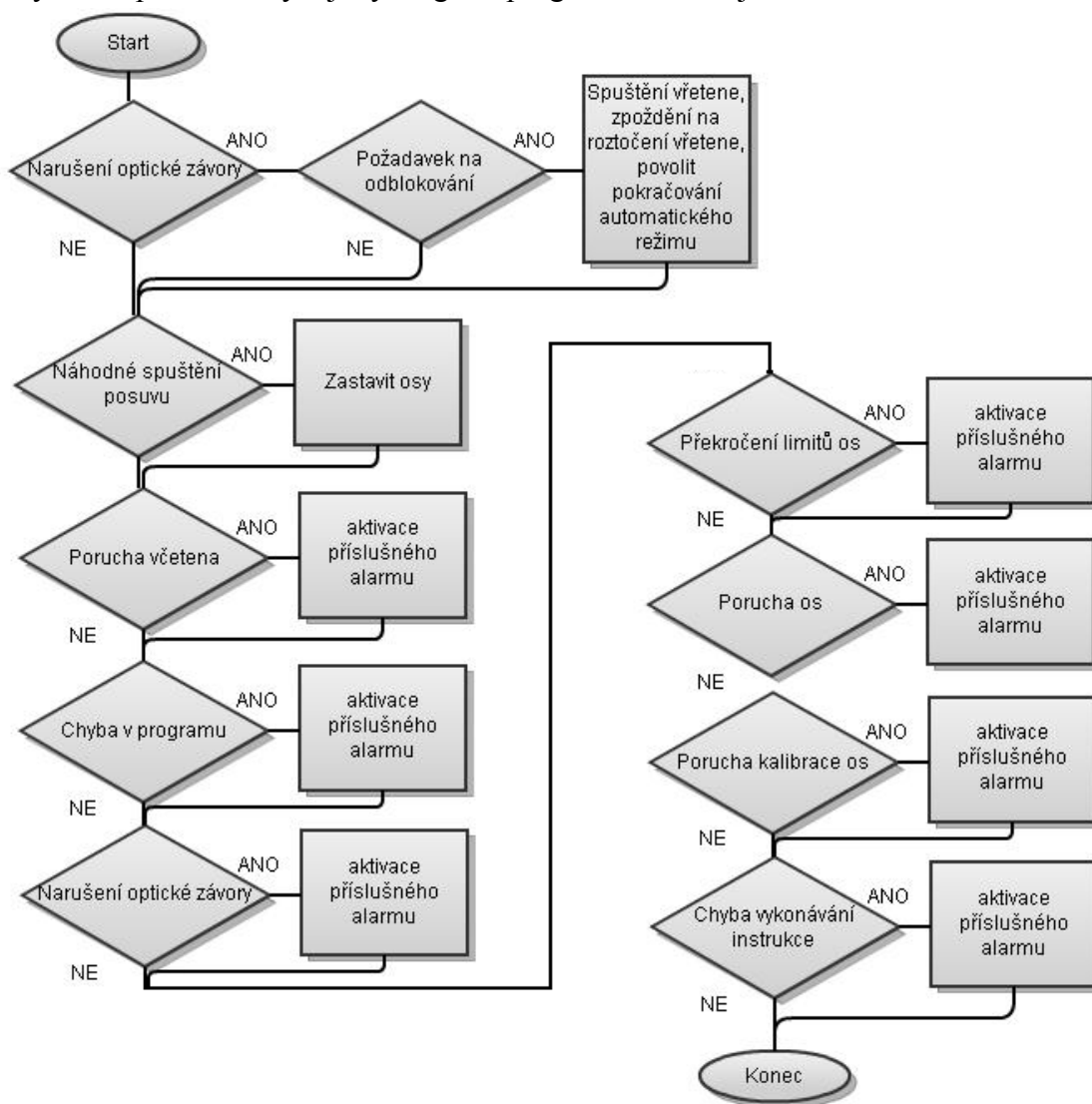


Obrázek č. 7.12 – Automatický režim

### 7.3.6 Kontrola chyb

Obsluha chybových stavů stroje se volá vždy na konci cyklu automatu, protože tuto část kódu vykonává procesor jako poslední, a tak nemůže dojít k přepsání důležitých hodnot jinými programy. Zde se provádí řešení narušení optické závory a aktivace jednotlivých alarmů.

Po narušení optické závory se automatický režim uvede do stavu, odkud se dá pokračovat pouze po stisknutí tlačítka pro pokračování. Program v tomto stavu ihned zastaví všechny osy. A jakmile se přestanou pohybovat, vypne i rotaci vřetene. Při opuštění tohoto stavu se zapne rotace vřetene a po době nutné k jeho roztočení se dokončí přerušená instrukce. Tuto dobu určuje parametr P instrukce G04. Další stav systému ošetřený v této rutině se týká nechtěného pohybu os nebo vřetene. Pokud není požadavek na ruční nebo automatický režim, je volána instrukce zastavující veškerý pohyb manipulátoru. Vývojový diagram programu zobrazuje obrázek č.: 7.13.



Obrázek č. 7.13 – Kontrola chyb

Indikované alarmové stavy se přenáší do vizualizace a jsou zobrazeny uživateli. První hlídaný stav se týká poruchy vřetene. Jedná se o stav, kdy nastal požadavek na rotaci vřetene, ale k jeho rotaci nedošlo. A druhý, kdy rotace vřetene neustala po skončení jejího požadavku. Pokud při kontrole instrukcí nalezne program chybnou instrukci, aktivuje alarm s názvem chyba při kontrole programu. Na rušení optické závory kromě okamžitého zastavení pohybu vyvolá také alarmové hlášení. V okamžiku, kdy se některá z os dostane mimo svůj rozsah, zobrazí se hlášení s názvem osy, která jej vyvolala. Příslušné hlášení vyvolá také chybový stav kterékoliv osy systému. Alarmy ohlašují i vzniklý problém při kalibraci osy. Poslední alarm nastane při chybném vykonání instrukce pohybu. Pro případ aktivace alarmu se v programu nachází i podprogram pro jejich restartování.

### **7.3.7 Program v safety procesoru**

Jak jsem již v kapitole popisující komponenty manipulátoru zmiňoval, kromě procesoru vykonávajícího uživatelský program se zde nachází i procesor pro bezpečnostní funkce aplikace. V našem případě se safety procesor kromě kontroly uživatelského procesoru stará také o vyčítání hodnot z bezpečnostní periferie připojené pomocí sběrnice DeviceNet, jejich zpracování a předávání do druhého procesoru. K bezpečnostní periférii je připojena optická závora a STOP tlačítko na rozváděči se servoměniči.

## **7.4 Vizualizace [47]**

Rozhraní člověk - stroj je jeden ze základních prvků každého systému, který může člověk sledovat nebo i ovládat. Toto rozhraní se v průmyslu využívá pro shromáždění důležitých informací o procesu, jejich přehledné prezentaci a v neposlední řadě i zadávání parametrů a řízení aplikace. Obsluha tak disponuje všemi potřebnými údaji a může zasahovat do chodu systému. V minulosti byly informace zobrazovány nejrůznějšími analogovými zobrazovači, ale v dnešních dobách se setkáme většinou s dotykovými LCD displeji, které kombinují zobrazovací a ovládací funkci.

Firma Allen Bradley nabízí nástroj pro vytváření vizualizací pomocí programu jménem Factory Talk View Studio. Pomocí tohoto nástroje se vytváří vizualizace s mnoha možnostmi od jednoduché indikace stavů procesu až po složité animované objekty měnící polohu nebo barvu a mnoho dalšího. V levé části nalezneme strom s jednotlivými komponenty, tyto komponenty poté umístíme na vytvořené obrazovky a přiřazujeme jim proměnné z PLC.

### 7.4.1 Úvodní stránka

Na úvodní stránce (obr. č.: 7.14) nalezneme název, autora a rok této diplomové práce. Nachází se zde také obrázek manipulátoru. Nejdůležitější část této stránky se nachází vlevo a jedná se o návod s tlačítky pro uvedení frézky do chodu. Jednotlivé texty budou mizet podle aktuálního stavu systému. Po stisknutí tlačítka „Reset os“ se osy restartují a uvedou do stavu připraveného k jejich zapnutí. Zmizí nápis „Restartujte OSY“. Následuje zapnutí os. Zapnutá osa deaktivuje svou brzdu, nabudí motor a pevně udržuje polohu osy. Zmizí nápis „Zapněte OSY“. Kalibrace os se provede vynulováním měřítek při aktivaci čidel určujících nulovou polohu. Po kalibraci zmizí nápis „Zkalibrujte OSY“. Pokud se všechny předchozí operace podařily bez chyby, zobrazí se zelený obrázek s nápisem „Frézka připravena“. Vypnutí os odbudí motory a aktivuje jejich brzdy. Ve spodní části nalezneme sadu tlačítek umožňující přechod mezi jednotlivými obrazovkami. Tyto tlačítka se nachází na všech obrazovkách. Poslední tlačítko této obrazovky s názvem „Konec“ ukončí vizualizaci.

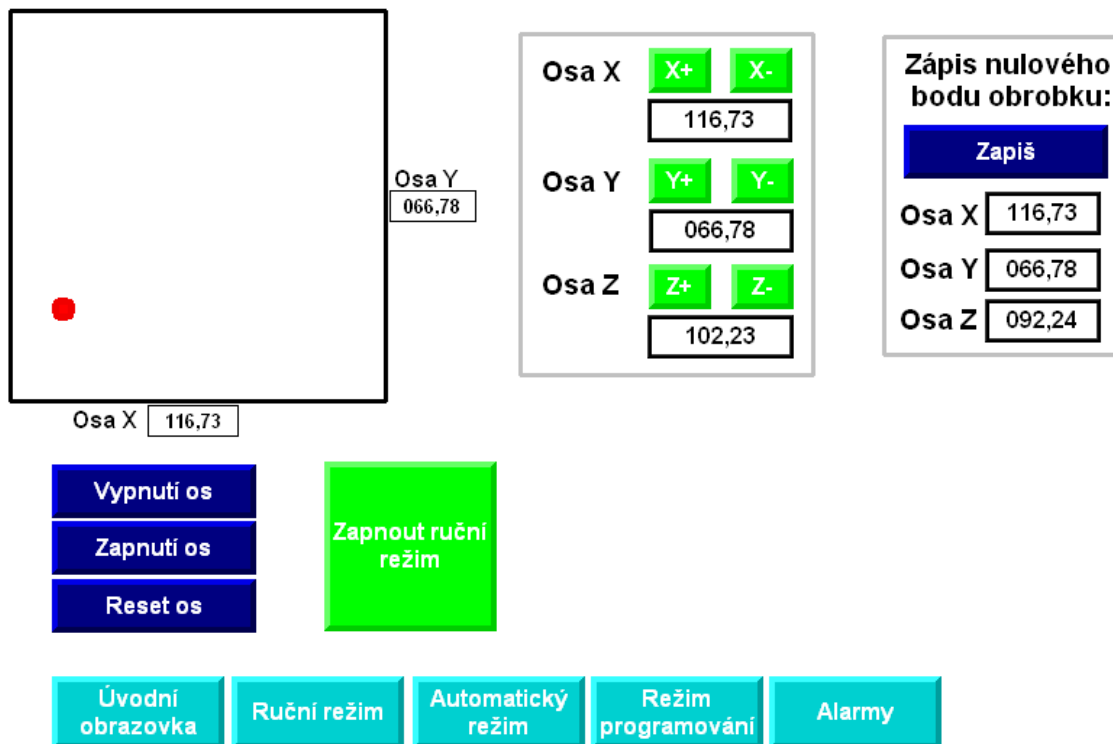


Obrázek č. 7.14 – Úvodní obrazovka

### 7.4.2 Ruční režim

V levé horní části vidíme rámeček znázorňující pracovní plochu manipulátoru a aktuální polohu vřetene. Napravo od tohoto ukazatele se nachází šestice tlačítek umožňující ruční posuv vřetene. Tlačítkem „Zapiš“ umístěném zcela vpravo nastavíme nulový bod obrobku na aktuální pozici nástroje. Hodnoty os jsou zobrazeny

pod tlačítkem. Následující dvě funkce se aktivují až po zapnutí ručního režimu (Tlačítko „Zapnout ruční režim“). Dále zde vidíme prvky pro přechod mezi obrazovkami a prvky ovládání stavu os jako na předchozí obrazovce.



Obrázek č. 7.15 – Obrazovka ručního režimu

### 7.4.3 Programovací režim

Tato obrazovka (obr. č.: 7.16) slouží pro nahrání programu do PLC, jeho prohlížení a kontrole. Povolení nahrávání programu spustíme tlačítkem „Zapnout režim programování“. V této chvíli se odblokuje paměť PLC a pomocí přiloženého dokumentu MS Excel lze nahrát instrukce do PLC. Stručný návod s obrázkem se nachází vpravo. Po dokončení nahrávání je potřeba program zkontrolovat tlačítkem „Zkontrolovat program“. Pokud proběhla kontrola úspěšně, zobrazí se hlášení v podobě zeleného obrázku a nápisu „Program OK“. Na této obrazovce může také obsluha nastavit zvětšení pro osy X a Y. Listování instrukcemi provádíme tlačítky „Předchozí“, „Další“ a „Pořadí instrukce“. Nahlížená instrukce se zobrazí v přílehlém políčku.



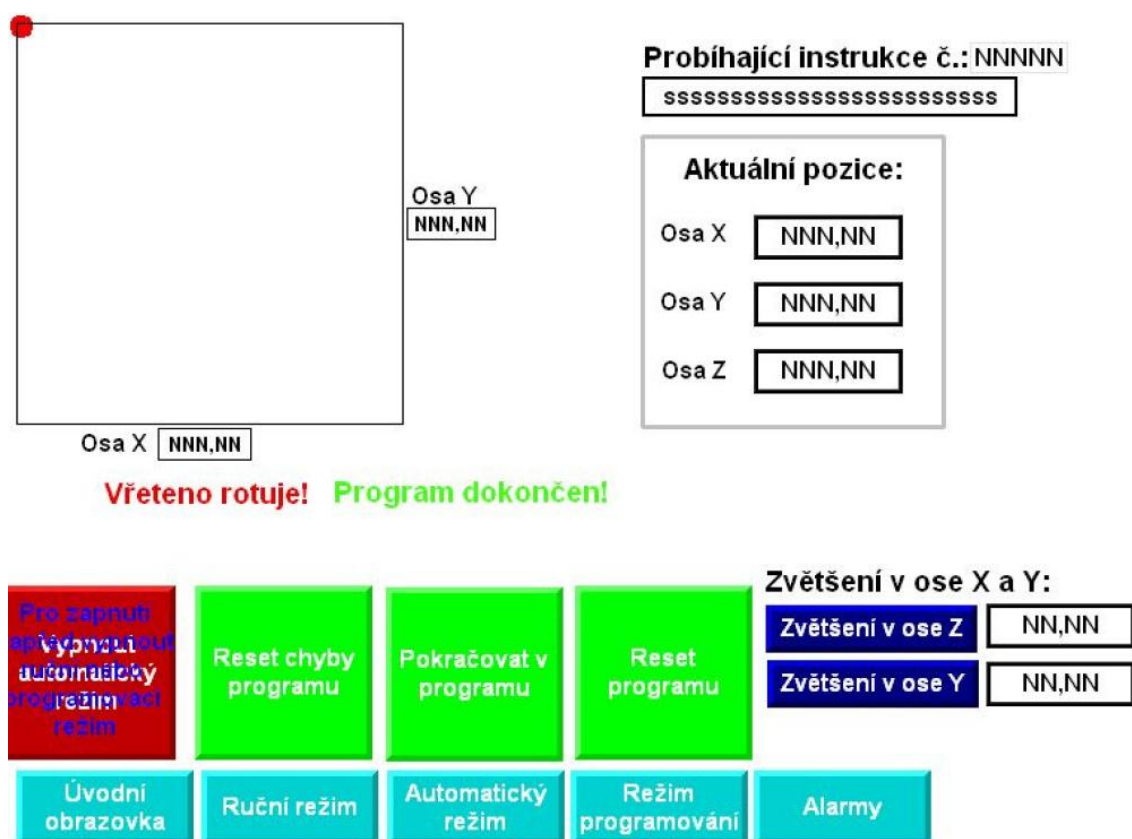
Obrázek č. 7.16 – Obrazovka programovacího režimu

## 7.4.4 Automatický režim

V automatickém režimu (obr. č.: 7.17) vidíme opět okno s aktuální pozicí nástroje, toto okno je doplněno o hlášení rotace vřetene a konce programu. Vpravo se zobrazuje aktuální vykonávaná instrukce a aktuální pozice nástroje ve všech osách. Před zapnutím automatického režimu lze také ještě upravit zvětšení os X a Y. Pro ovládání tohoto režimu se zde nachází tlačítka Vypnutí/zapnutí automatického režimu, reset chyby vzniklé narušením optické závory, pozastavení vykonávání programu a reset programu po jeho selhání nebo po jeho zopakování.

## 7.4.5 Alarmy

Poslední obrazovka (obr. č.: 7.18) naší vizualizace zobrazuje vzniklé alarmy. Pod zobrazovačem alarmů se nachází tlačítka pro aktualizaci, dočasnou deaktivaci a smazání vzniklých alarmů. Tlačítkem „Close“ se okno zavře a zobrazí se předchozí obrazovka.



Obrázek č. 7.17 – Obrazovka automatického režimu



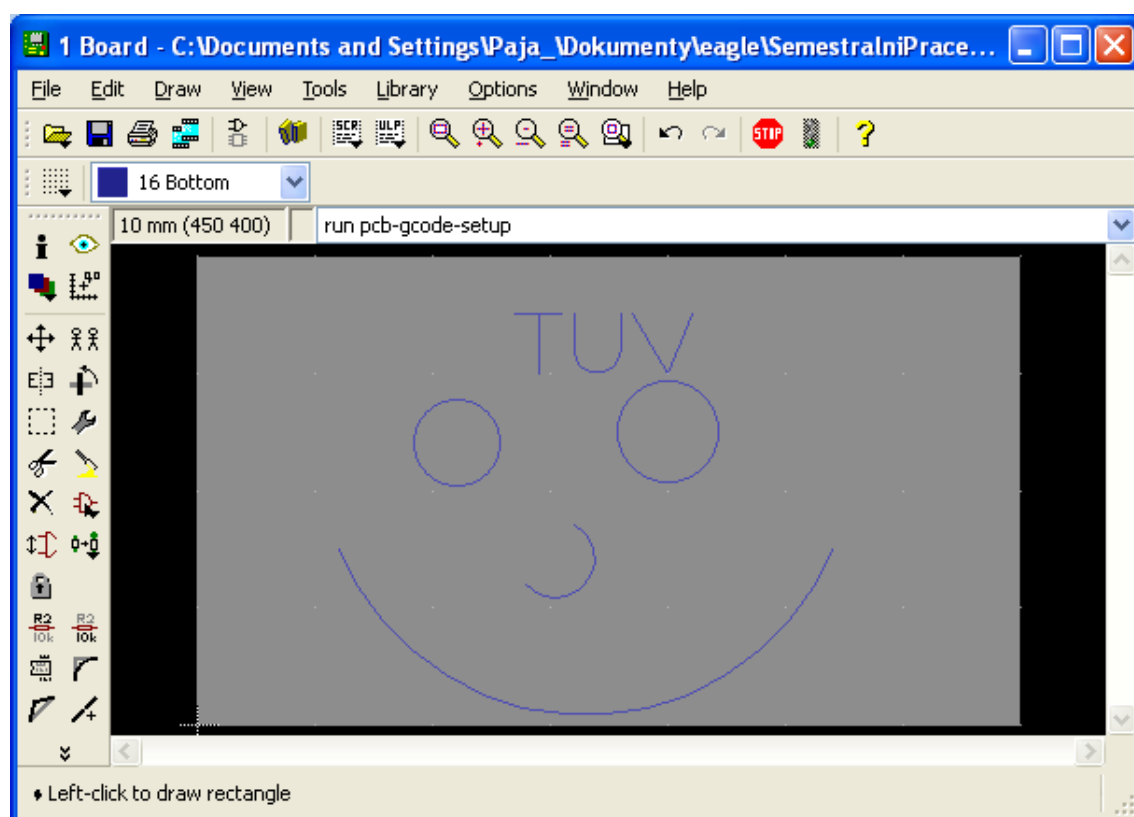
Obrázek č. 7.18 – Obrazovka s alarmy

## 8 UKÁZKOVÝ PROGRAM

V předchozích kapitolách jsem popisoval návody na nakreslení obrázku, vygenerování souboru s kódem, přesun kódu do PLC a výsledné vykreslení. Nyní si postup ukážeme na konkrétním příkladě, který si projdeme od začátku až do konce.

### 8.1 Prostředí Eagle

Po nainstalování programu Eagle si do jeho adresáře do podsložky s názvem „upl“ zkopírujeme extrahovaný obsah archivu s programem PCB-GCODE. Nyní již můžeme přistoupit ke spuštění Eaglu. Pomocí menu „File“ vytvoříme nový projekt a pojmenujeme jej. V tomto projektu si vytvoříme soubor s deskou plošných spojů (DPS) a také pojmenujeme. Dvojím poklikáním na vytvořený soubor s DPS se otevře okno pro její vytváření. Zde si v levé části okna vybereme nástroj s názvem „Rect“, a poté mu zvolíme parametr „bPlace“. Vybraným nástrojem zadáme programu pracovní plochu, ve které se budeme pohybovat. Na této ploše si pomocí nástrojů „Wire“, „Circle“ a „Arc“ nakreslíme požadovaný obrázek. Musíme si dát pozor na to, abychom obrázek kreslili ve vrstvě „Bottom“. Po nakreslení projekt uložíme a zadáme do příkazového řádku „run pcb-gcode-setup“. (obr. č.: 8.1)

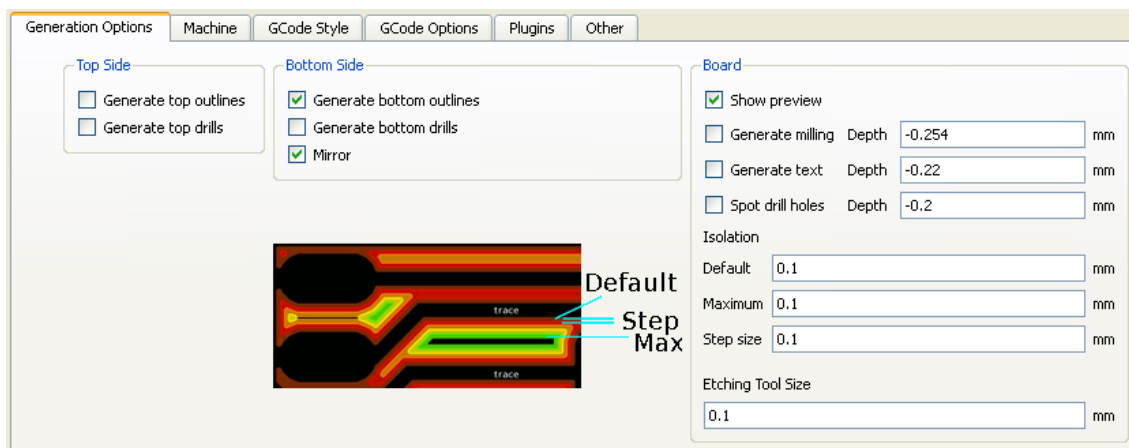


Obrázek č. 8.1 – Tvorba obrázku v Eaglu



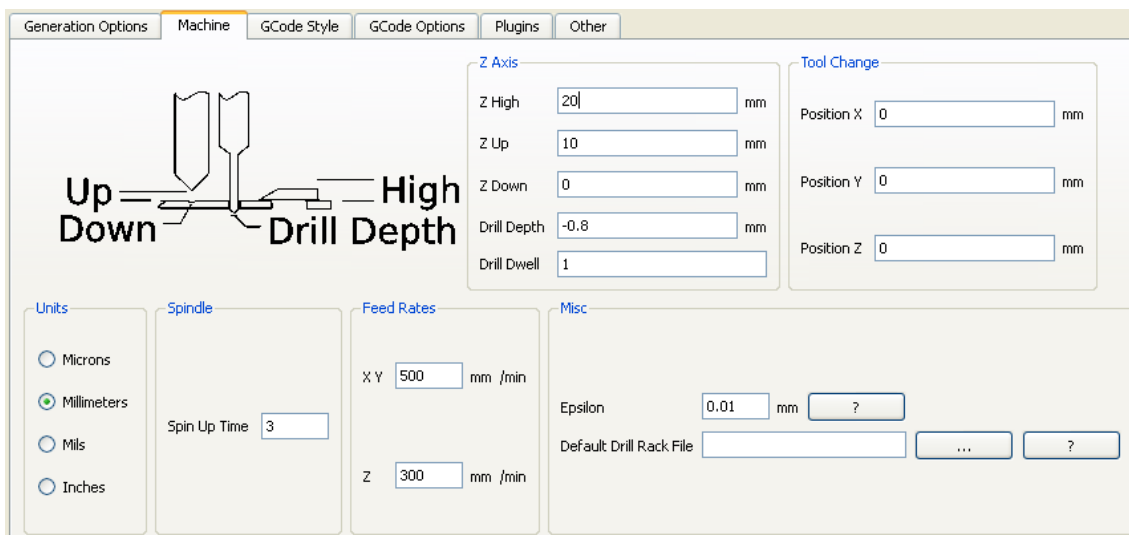
## 8.2 Nastavení PCB-GCODE

Po zadání příkazu „run pcb-gcode-setup“ se zobrazí prostředí, kde nastavíme potřebné parametry pro správné vygenerování kódu. Na první záložce nastavíme parametry „Generate bottom outlines“ a „Mirror“. Hodnotu parametrů „Default“, „Maximum“ a „Step size“ nastavíme na desetinu milimetru.(viz. obr. č.: 8.2)



Obrázek č. 8.2 – Nastavení PCB-GCODE I.

V následující záložce nastavíme jednotky na milimetry, dobu potřebnou k ustálení vřetene (spindle) na 3 s, rychlosti pohybů (Feed Rates) a polohy osy Z podle následujícího obrázku.



Obrázek č. 8.3 – Nastavení PCB-GCODE II.

V záložce „GCode Options“ (obr. č.: 8.4) odznačíme všechna políčka, abychom negenerovali více souborů, než je potřeba a stisknutím tlačítka „Accept and make my board“ vygenerujeme soubor s instrukcemi. Tento soubor nalezneme ve složce projektu.

The screenshot shows the 'GCode Options' tab in the PCB-GCODE III software. It includes sections for 'NC File Comments', 'Other Options', 'File Naming', 'Top (Component) Side Files', and 'Bottom (Solder) Side Files'. The 'File Naming' section is particularly detailed, showing fields for 'Filename Base', 'Extension', and words for various operations like 'etch', 'drill', 'mill', 'text', 'top', and 'bottom'. The 'Side Files' sections provide templates for different operations, using placeholders like \$.SIDE, \$.FILE, and \$.EXT.

Obrázek č. 8.4 – Natavení PCB-GCODE III.

## 8.3 Nastavení manipulátoru

Po zapnutí PLC manipulátoru, rozváděče se servoměniči a nahrání programu do PLC spustíme vizualizaci v programu Factory Talk View Studio. Podle popisu v kapitole 7.4 spustíme a kalibrujeme osy. Přepneme se do ručního režimu, kde nastavíme nulový bod. V našem případě se jedná o počátek papíru umístěného na stole manipulátoru. Po jeho nastavení aktivujeme režim nahrávání programu a spustíme aplikaci pro nahrávání kódu do PLC (dokument MS Excel). Zde postupujeme podle kapitoly 6.3, která popisuje postup nahrávání kódu do PLC. (viz. obr. č.: 8.5)

B	C	D	E	F
<b>Zadejte cestu k souboru!</b>				G21
<b>Cesta k souboru:</b>	F:\SMAJLIK.bot.etch.txt			(Absolute Coordinates)
<b>Vybrat cestu k souboru</b>				G90
<b>Načteno řádků:</b>	576			G00 X0.0000 Y0.0000
				M03
<b>Velikost souboru [b]:</b>	14681			G04 P3.000000
				G00 Z10.0000
<b>Načti soub</b>				G00 X442.0732 Y250.0000
				G01 Z0.0000 F300.00
Naposled načteno v:	2:33:55 odp.			G01 X441.8827 Y246.0007 F500.00
				G01 X441.3129 Y242.0376
<b>Smazat</b>				G01 X440.3689 Y238.1466
				G01 X439.0594 Y234.3629
<b>Zapiš</b>				G01 X437.3961 Y230.7209
				G01 X435.3942 Y227.2535
				G01 X433.0717 Y223.9920

Obrázek č. 8.5 – Nahrávání kódu do PLC

Po úspěšném nahrání kódu vypneme programovací režim a tlačítkem „Kontrola programu“ spustíme kontrolu nahraného kódu. Pokud kontrola neodhalila chyby, můžeme přistoupit k automatickému vykonávání kódu. Nejprve ale musíme nastavit zmenšení v osách X a Y na hodnotu 0,2. Proto abychom velikost obrázku přizpůsobili velikosti formátu A4. Nyní již zapneme automatický režim a manipulátor vykreslí požadovaný obrázek. Tento náčrt přikládám jako přílohu č. 1. (viz. obr. č.: 8.6)



**Obrázek č. 8.6 – Vykreslení obrázku**

## 9 ZÁVĚR

Hlavním cílem této práce se stal vývoj programového vybavení umožňující použití modelu manipulátoru v laboratoři jako CNC gravírovací stroj. Kromě tvorby programů se zabývá práce i návrhem přestavby potřebných částí pro jejich další funkci a bezpečnost.

V první části této práce jsem se zaměřil na obecný popis CNC strojů hlavně CNC frézek. Nalezneme zde základní rozdělení CNC strojů a popis jejich programování na nejnižší úrovni. Následuje rozdělení CNC frézek, popis jejich konstrukce a popis dílčích systémů prodáváných strojů.

Následující kapitola popisuje model manipulátoru od firem Robotech a Zámečnictví - MSK a jeho jednotlivé řídicí komponenty, které dodala firma Allen Bradley. Hlavním „mozkem“ tohoto systému se stalo PLC s procesorovým párem GuardLogix obsahujícím procesor pro řízení a koprocessor pro správu bezpečnostních prvků. PLC komunikuje se servoměniči Kinetix 2000 pomocí optického rozhraní Sercos. Systém Kinetix 2000 podporuje řízení až osmi os a umožňuje jejich koordinovaný pohyb, potřebný k použití manipulátoru pro obrábění.

Následuje část uvádějící návrh konstrukčních úprav nezbytných k proměně manipulátoru v CNC frézku. Úpravy se týkají namontování ochranných krytů pro zabránění poranění obsluhy a znečištění prostředí. Pro zabezpečení obráběného předmětu se zmiňují i o možnostech jeho upnutí na obráběcí desce. Následně se zde zmiňují o řešení problémů s upínáním nástroje a jeho rotací, který byl vyřešen zakoupením ruční frézky. V této kapitole se také rozebírá řešení automatického spouštění frézky pomocí PLC, safety I/O modulu a instalovaného stykače. Protože na modelu manipulátoru pracuje více studentů, nebylo možné většinu navrhovaných úprav realizovat. Částečně se podařilo vyřešit rotaci, upínání a spouštění nástroje. Pro neinvazivní řešení navrhuji vytvořit upínací přípravek na zakoupenou frézku, který by bylo možno pomocí několika šroubů snadno demontovat.

Kapitola věnovaná generování G-kódu se zabývá popisem vybraného programu pro generování G-kódu z nakreslených obrázků v prostředí programu Eagle. Podrobně zde popisují jednotlivé kroky instalace programu a jeho inicializace s detailním popisem jednotlivých možností, které jsem pro přehlednost doplnil obrázky. Hlavní parametry programu PCB-GCODE se týkají jednak oblasti, která se má převádět na kód, ale také nastavení parametrů frézování a stylu tvorby kódu. Uvedl jsem část vygenerovaného kódu a popsal, jaký příkaz se pod jednotlivými instrukcemi skrývá. Závěrem kapitoly popisují ještě dvojici programů pro převod obrázku na G-kód. Tyto programy podporují také odstupňování hloubky frézování v závislosti na jasu obrázku, což v praxi znamená věrohodnější výsledek gravírování.

Kapitola s názvem Přenos dat do PLC pojednává o transportu instrukcí z vygenerovaného souboru do paměti PLC, kde se následně použijí jako příkazy pro její ovládání. Úvodem se zde nachází stručný popis DDE serveru použitého pro přenos dat. Následuje popis nastavení tohoto serveru pro naši konkrétní aplikaci, který byl vybrán

díky podpoře programu MS Excel. Druhá polovina kapitoly se zaměřuje na popis vytvořené aplikace, zejména na úsek sloužící pro komunikaci s DDE serverem.

Úvod kapitoly popisující vytvořený program v PLC je věnován popisu speciálních instrukcí, které slouží pro řízení servoměničů. Instrukce se týkají nejen pohybu os, ale také jejich kalibrace nebo zapínání a vypínání. Vytvořený program je tvořen hlavní smyčkou, ze které se následně volají potřebné podprogramy. Jednotlivé podprogramy řeší ovládání manipulátoru v ručním či automatickém režimu nebo se starají o dekodování instrukcí a chybové stavy. Program po svém spuštění dovolí uživateli aktivovat osy a umožní jejich ruční posuv s možností zadání nulového bodu obrobku. Po nahrání programu z vytvořené aplikace musí následovat kontrola, při které se zjistí případné chyby v nahraném kódu. Je-li kód v pořádku, zpřístupní se uživateli automatický režim, ve kterém se kód vykonává nezávisle na obsluze. Při tvorbě programu byl kladen důraz na bezpečnost osob obsluhy a omezení možnosti vzniku chyb vedoucích k poškození zařízení.

Poslední část řetězce programového vybavení tvoří vizualizace. Ta umožňuje uživateli sledovat nejrůznější stavy systému a reagovat na ně. Na úvodní obrazovce nalezneme funkce pro aktivaci/deaktivaci, kalibraci a restartování os. Následují obrazovky pro ovládání automatického a ručního režimu. Nahrávání programu do PLC je nutné povolit na obrazovce programovacího režimu. Za nedílnou součástí vizualizace se v dnešní době považuje hlášení chybových stavů, a proto i v našem systému nechybí okno, kde se zobrazují tato hlášení.

Závěrečná kapitola se věnuje popisu ukázky celého systému od instalace programů až po zhotovení nakresleného vzoru v automatickém režimu. Pro větší srozumitelnost popis doplňují obrázky nejdůležitějších kroků.

Výsledkem této diplomové práce se stala CNC frézka vytvořená z manipulátoru v laboratoři. Tato frézka není po konstrukční stránce zcela dokončena z důvodů práce ostatních studentů na tomto zařízení. Po programové stránce umožňuje uživateli importovat předem vytvořený kód do své paměti a jeho následné vykonávání. Importovaný kód smí obsahovat instrukce G21(absolutní programování), G90 (programování v mm), G00 (rychloposuv), G01 (lineární interpolace), G02 a G03 (kruhová interpolace) G04 (časová prodleva), M03 (zapnutí rotace vřetene), M05 (vypnutí rotace vřetene) a M02 (konec programu) včetně parametrů rychlosti, doby zpoždění či velikosti poloměru. Základní nedostatky v kódu odhalí naprogramovaná kontrola kódu. Frézka je také naprogramována i pro definici nulového bodu obrobku nebo pro změnu měřítka os X a Y. Program zabráni spuštění režimu, pokud to není bezpečné. Při narušení optické závory se aplikace zastaví a vyčká na pokyn obsluhy. Přesnost polohování byla vyčíslena na 0,01mm a maximální doporučená rychlost posuvu os na 500mm/min.

Jako pokračování v tomto projektu navrhuji dokončení konstrukčního přizpůsobení manipulátoru pro jeho další využití jako CNC frézky. Programovou část navrhuji rozšířit o možnost zpracovávání kompaktního G-kódu a následně další přidávání instrukcí, které manipulátor dekoduje a vykoná.

# Literatura

- [1] BORDOVSKÝ, L. 2009. CNC fréza – návrh a realizace. VOŠ a SPŠE Olomouc.  
[online] [cit. 2011-04-26]. Dostupné z:  
<http://luda.zaridi.to/download/AP.pdf>
- [2] ING. CHUDOBA, M. Základy programování a obsluha CNC strojů.  
[online] [cit. 2011-04-26]. Dostupné z:  
[http://www2.sps-jia.cz/~hill/zakl\\_prog.pdf](http://www2.sps-jia.cz/~hill/zakl_prog.pdf)
- [3] G-KÓD [online] [cit. 2011-04-26]. Dostupné z:  
<http://homel.vsb.cz/~moz017/G-code/>
- [4] TKÁČOVÁ, A. *Programování CNC strojů se systémem fanuc*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2009. 32s., 2s. příloh Vedoucí bakalářské práce Ing. Milan Kalivoda.  
[online] [cit. 2011-04-26]. Dostupné z:  
[http://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=17122](http://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=17122)
- [5] Výrobky firmy Trimill  
[online] [cit. 2011-04-26]. Dostupné z:  
[www.trimill.cz](http://www.trimill.cz)
- [6] Hlava od firmy Renishaw [online] [cit. 2011-04-26]. Dostupné z:  
<http://coordinate-measuring-machine.net/wp-content/uploads/2011/01/Renishaw-PH20.jpg>
- [7] Otočný stůl od firmy Fermat [online] [cit. 2011-04-26]. Dostupné z:  
<http://www.fermatmachinery.com/images/1318-Otocny-stul.jpg>
- [8] Upínky [online] [cit. 2011-04-26]. Dostupné z:  
[http://www.poziadavka.sk/img\\_users/ponuky/full/51610\\_2.gif](http://www.poziadavka.sk/img_users/ponuky/full/51610_2.gif)
- [9] GM Electronic spol. s.r.o.[online] [cit. 2011-12-16]. Dostupné z:  
<http://www.gme.cz/mikrovrtacky-a-frezky/vrtaci-bruska-dremel-dremel-300-series-1-55-p732-977/>
- [10] Frézy [online]. [cit. 2011-04-26]. Dostupné z:  
<http://www.logismarket.cz/ip/kaestner-frezy-se-stopkou-frezy-se-stopkou-431501-FGR.jpg> a [http://www.mmspektrum.com/multimedia/image/88/8829\\_big.jpg](http://www.mmspektrum.com/multimedia/image/88/8829_big.jpg)

- [11] Průmyslový Ethernet IV: Principy průmyslového Ethernetu [online]. [cit. 2011-04-26]. Dostupné z: [http://www.odbornecasopisy.cz/index.php?id\\_document=34198](http://www.odbornecasopisy.cz/index.php?id_document=34198)
- [12] GOLIS, Š. *Řízení modelu manipulátoru*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 36 s. Vedúci bakalárskej práce Ing. Radek Štohl, Ph.D.
- [13] Matějček, Miroslav. *Řízení 3 os modelu manipulátoru*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. s. 61. Ing. Radek Štohl Ph.D.
- [14] ControlTech s. r. o. [online]. [cit. 2011-04-26]. Dostupné z: <http://www.controltech.cz/products/sercos.php?pid=2>
- [15] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z: <http://ab.rockwellautomation.com/programmable-controllers/controllogix>
- [16] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z: <http://ab.rockwellautomation.com/programmable-controllers/guardlogix-controllers-with-safety>
- [17] PLC Hardware. [online]. [cit. 2011-04-26]. Dostupné z: <http://www.plchardware.com/Products/84/5398806/5398809/RA-1756-LSP-B-UPP.aspx> a <http://www.plchardware.com/Products/RA-1756-L62S.aspx>
- [18] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z: <http://www.ab.com/en/epub/catalogs/12762/2181376/2416247/360807/>
- [19] PLC Hardware. [online]. [cit. 2011-26-04]. Dostupné z: <HTTP://WWW.PLCHARDWARE.COM/PRODUCTS/SPECIALOFFER/RA-1756-L61-B-UPP.ASPX>
- [20] DeviceNet Introduction. [online]. [cit. 2011-26-04]. Dostupné z: <http://www.rtaautomation.com/devicenet/>
- [21] ControlTech. [online]. [cit. 2011-26-04]. Dostupné z: <http://www.controltech.cz/products/devicenet.php?pid=2>
- [22] PLC Hardware. [online]. [cit. 2011-26-04]. Dostupné z: <http://www.plchardware.com/Products/RA-1756-DNB-A-UPP.aspx>

- [23] Ing. Zdeněk Šibrava, Rozhraní SERCOS. *Automa*, r. 2003, č. 04. [online]. [cit. 2011-04-26]. Dostupné z:  
[http://www.odbornecasopisy.cz/index.php?id\\_document=28780](http://www.odbornecasopisy.cz/index.php?id_document=28780)
- [24] ControlTech. [online]. [cit. 2011-26-04]. Dostupné z:  
<http://www.controltech.cz/products/kinetix2000.php?pid=2>
- [25] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z:  
<http://ab.rockwellautomation.com/Motion-Control/Kinetix-2000-Servo-Drive>
- [26] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z:  
[http://samplecode.rockwellautomation.com/idc/groups/literature/documents/um/2093-um001\\_-en-p.pdf](http://samplecode.rockwellautomation.com/idc/groups/literature/documents/um/2093-um001_-en-p.pdf)
- [27] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z:  
<http://ab.rockwellautomation.com/motion-control/mp-series-servo-motors/low-inertia>
- [28] Adate s.r.o. [online]. [cit. 2011-04-26]. Dostupné z:  
<http://www.adate.cz/upinani-obrobku/>
- [29] BOST SK, a.s.[online]. [cit. 2011-04-26]. Dostupné z:  
<http://www.bostap.eu/sk/produkty/bsac-600-850-1100>
- [30] Logix 5000 Controllers Motion Instruction - User Manual. *Rockwell Automation-literature*. 2008 .[online] [cit. 2011-04-26]. Dostupné z:  
[http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/1756-rm007\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/1756-rm007_-en-p.pdf)
- [31] Brusselsprout.[online]. [cit. 2011-12-16]. Dostupné z:  
<http://www.brusselsprout.org/PCB-Routing/>
- [32] Phorum.[online]. [cit. 2011-12-16]. Dostupné z:  
<http://www.pcbgcode.org/>
- [33] CadSoft Computer GmbH.[online]. [cit. 2011-12-16]. Dostupné z:  
<http://www.eagle.cz/>
- [34] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z:  
<http://www.ab.com/en/epub/catalogs/3377539/5866177/5985760/6388285/5985790/6392754/print.html>



- [35] Rockwell Automation, Inc. [online]. [cit. 2011-04-26]. Dostupné z:  
<http://www.ab.com/en/epub/catalogs/12762/2181376/2416247/360807/>
- [36] ROH, M. Praha: České vysoké učení technické v Praze, Fakulta elektrotechnická, 2007. 110s. Vedoucí diplomové práce Ing. Jindich Fuka. [online]. [cit. 2010-25-05]. Dostupné z:  
[http://dce.felk.cvut.cz/dolezilкова/diplomky/2007/dp\\_2007\\_roh\\_michal/dp\\_2007\\_Roh\\_michal.pdf](http://dce.felk.cvut.cz/dolezilкова/diplomky/2007/dp_2007_roh_michal/dp_2007_Roh_michal.pdf)
- [37] Ing. Bořek Vích, ControlLogix – řídicí systém budoucnosti. Automa, r. 2000, č. 06 [online]. [cit. 2010-25-05]. Dostupné z:  
[http://www.odbornecasopisy.cz/index.php?id\\_document=27740](http://www.odbornecasopisy.cz/index.php?id_document=27740)
- [38] Strojnet.cz Milan Sigmund [online]. [cit. 2012-05-08]. Dostupné z:  
<http://www.strojnet.cz/clanky/obrabeci-stroje-cnc.php>
- [39] www.C-N-C.cz [online]. [cit. 2012-05-08]. Dostupné z:  
<http://www.c-n-c.cz/>
- [40] IngPetr Mikšovský, ControlTech, s. r. o., Průmyslové sítě pro všechny sféry automatizace. Automa, r. 2001, č. 11. [online]. [cit. 2012-05-08]. Dostupné z:  
[http://www.odbornecasopisy.cz/index.php?id\\_document=33716](http://www.odbornecasopisy.cz/index.php?id_document=33716)
- [41] Abraham Valdes [online]. [cit. 2012-05-08]. Dostupné z:  
<http://www.imagetogcode.com/>
- [42] Carken Co. [online]. [cit. 2012-05-08]. Dostupné z:  
[www.deskam.com](http://www.deskam.com)
- [43] Jarosław Mazurek [online]. [cit. 2012-05-08]. Dostupné z:  
<http://www.inf-el.com/cnc/cnc.htm>
- [44] Software toolbox, Inc. [online]. [cit. 2012-05-08]. Dostupné z:  
[http://www.softwaretoolbox.com/tech\\_support/techexpertisecenter/objectsoftware/ddewhatisit/ddewhatisit.html](http://www.softwaretoolbox.com/tech_support/techexpertisecenter/objectsoftware/ddewhatisit/ddewhatisit.html)
- [45] Computing.net [online]. [cit. 2012-05-08]. Dostupné z:  
<http://www.computing.net/answers/windows-xp/dde-server/173909.html>
- [46] PLCdev [online]. [cit. 2012-05-08]. Dostupné z:  
[http://www.plcdev.com/connecting\\_controllogix\\_excel](http://www.plcdev.com/connecting_controllogix_excel)

[47] Rockwell Automation, Inc. [online]. [cit. 2012-05-15]. Dostupné z:  
[http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/ftae-rm001\\_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/rm/ftae-rm001_-en-e.pdf)

## Seznam obrázků

Obrázek č. 2.1 - Vertikální a horizontální vřeteno [5] .....	15
Obrázek č. 2.2 – Souřadný systém [2] .....	15
Obrázek č. 2.3 – Osy A a B [6] .....	16
Obrázek č. 2.4 – Výměnné břity [10] .....	17
Obrázek č. 3.1 – Model manipulátoru .....	18
Obrázek č. 3.2 – Schéma sítě Ethernet/IP [11] .....	19
Obrázek č. 3.3 – Procesory PLC [17] .....	20
Obrázek č. 3.4 – Ethernet/IP modul [19] .....	21
Obrázek č. 3.5 – DeviceNet modul [22] .....	22
Obrázek č. 3.6 – Kinetix 2000 [24] .....	24
Obrázek č. 3.7 – MPL A310 [27] .....	25
Obrázek č. 4.1 – Ruční frézka [9] .....	28
Obrázek č. 4.2 – Zapojení stykače frézky [12] .....	29
Obrázek č. 5.1 – Prostředí Eagle .....	32
Obrázek č. 5.2 – Prostředí editoru desky .....	33
Obrázek č. 5.3 – Prostředí PCB GCODE I. ....	33
Obrázek č. 5.4 – Prostředí PCB GCODE II. ....	35
Obrázek č. 5.5 – Prostředí PCB GCODE III. ....	36
Obrázek č. 5.6 – Návrh desky .....	37
Obrázek č. 5.7 – Prostředí Image to G-code [41] .....	38
Obrázek č. 5.8 – Prostředí DeskEngrave [42] .....	39
Obrázek č. 5.9 – Prostředí B2G [43] .....	39
Obrázek č. 6.1 – Export tagu pro DDE server .....	40
Obrázek č. 6.2 – Uložení tagů pro DDE server .....	41
Obrázek č. 6.3 – Konfigurace DDE v RSLinx .....	41
Obrázek č. 6.4 – Založení DDE serveru .....	42
Obrázek č. 6.5 – Další konfigurace DDE serveru .....	42
Obrázek č. 6.6 – Prostředí pro zápis dat do PLC .....	43
Obrázek č. 7.1 – MASR [30] .....	46

Obrázek č. 7.2 – MSO [30] .....	46
Obrázek č. 7.3 – MAH [30] .....	47
Obrázek č. 7.4 – MCLM [30].....	48
Obrázek č. 7.5 – Parametr Direction [30] .....	49
Obrázek č. 7.6 – MCCM [30] .....	50
Obrázek č. 7.7 – MAJ [30].....	51
Obrázek č. 7.8 – MAS [30] .....	52
Obrázek č. 7.9 – MSF [30].....	52
Obrázek č. 7.10 – Dekódování instrukce .....	55
Obrázek č. 7.11 – Kontrola instrukcí .....	56
Obrázek č. 7.12 – Automatický režim .....	57
Obrázek č. 7.13 – Kontrola chyb .....	58
Obrázek č. 7.14 – Úvodní obrazovka.....	60
Obrázek č. 7.15 – Obrazovka ručního režimu.....	61
Obrázek č. 7.16 – Obrazovka programovacího režimu.....	62
Obrázek č. 7.17 – Obrazovka automatického režimu .....	63
Obrázek č. 7.18 – Obrazovka s alarmy .....	63
Obrázek č. 8.1 – Tvorba obrázku v Eaglu.....	64
Obrázek č. 8.2 – Natavení PCB-GCODE I. ....	65
Obrázek č. 8.3 – Natavení PCB-GCODE II.....	65
Obrázek č. 8.4 – Natavení PCB-GCODE III. ....	66
Obrázek č. 8.5 – Nahrávání kódu do PLC .....	66
Obrázek č. 8.6 – Vykreslení obrázku .....	67

## Seznam tabulek

Tabulka č. 2.1 – G-kód [2].....	13
Tabulka č. 2.2 – M-kód [2] .....	14
Tabulka č. 7.1 – Konfigurace PLC.....	45
Tabulka č. 7.2 – Parametr Termination Type [30].....	47
Tabulka č. 7.3 – Parametr Circle type [30] .....	49

## Seznam příloh

Příloha 1. Narýsovaný obrázek

Příloha 2. CD

## Seznam příloh na CD

Příloha 1. Dipolmová práce

Příloha 2. Program pro PLC

Příloha 3. Program vizualizace

Příloha 4. Program pro generování G-kódu

Příloha 5. Dokument MS Excel s makry pro zápis instrukcí

Příloha 6. Reporty programů

Příloha 7. Další obrázky

## Seznam zkratk

Mbps	Megabits per second
DPS	Deska plošných spojů
Ethernet/IP	Ethernet Industrial Portokol
DDE	Dynamic Data Exchange
OPC	OLE for Process Control
NC	Numerical Control
CNC	Computer Numerical Control
PLC	Programmable Logic Controller
SIL3	Safety integrity level