



TECHNICKÁ UNIVERZITA V LIBERCI  
Fakulta mechatroniky, informatiky  
a mezioborových studií ■

# Řízení víceosého servopohonu v prostředí LabVIEW

## Diplomová práce

*Studijní program:* N2612 – Elektrotechnika a informatika

*Studijní obor:* 3906T001 – Mechatronika

*Autor práce:* **Bc. Jiří Šlechta**

*Vedoucí práce:* doc. Ing. Milan Kolář, CSc.



## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jiří Šlechta**  
Osobní číslo: **M14000205**  
Studijní program: **N2612 Elektrotechnika a informatika**  
Studijní obor: **Mechatronika**  
Název tématu: **Řízení víceosého servopohonu v prostředí LabVIEW**  
Zadávací katedra: **Ústav mechatroniky a technické informatiky**

### Z á s a d y p r o v y p r a c o v á n í :

1. Seznamte se se způsobem řízení servopohonů v prostředí NI LabVIEW.
2. Prostudujte možnosti aplikování servopohonů NI a proveďte diskusi aplikování alternativně dalších servopohonů (např. Mitsubishi, OMRON, Siemens).
3. Proveďte návrh a diskusi modelové aplikace dvouosého servomechanizmu řízeného kartou typu NI 7342, při návrhu se zaměřte na implementaci do programového prostředí LabVIEW s cílem dosažení práce v reálném čase.
4. Odkoušejte navržené řešení při aplikování v modelové úloze a navrhněte rozšíření aplikace na víceosé řešení (např. řízení robota).


Rozsah grafických prací: **dle potřeby dokumentace**  
Rozsah pracovní zprávy: **cca 40–50 stran**  
Forma zpracování diplomové práce: **tištěná/elektronická**  
Seznam odborné literatury:

- [1] NI Motion Control, National Instruments. [online]. 2015 [cit. 2015-10-08].  
Dostupné z <http://www.ni.com/pdf/manuals/371242b.pdf>
- [2] Kretschmerová, L., Vlach, J.: Programování v LabVIEW v příkladech.  
Skriptum TUL, Liberec 2014, ISBN: 978-80-7372-167-2.

Vedoucí diplomové práce: **doc. Ing. Milan Kolář, CSc.**  
Ústav mechatroniky a technické informatiky  
Konzultant diplomové práce: **Ing. Jaroslav Vlach, Ph.D.**  
PRECIOSA a.s., Jablonec n. Nisou  
Datum zadání diplomové práce: **10. října 2015**  
Termín odevzdání diplomové práce: **16. května 2016**

  
prof. Ing. Václav Kopecký, CSc.  
děkan



  
doc. Ing. Milan Kolář, CSc.  
vedoucí ústavu

V Liberci dne 10. října 2015

## Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé diplomové práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 12.5.2016

Podpis: 

# Poděkování

Tímto bych chtěl poděkovat všem, kteří se na tvorbě této diplomové práce podíleli. Zvláště pak konzultantovi Ing. Jaroslavu Vlachovi, Ph.D. a vedoucímu Doc. Ing. Milanu Kolářovi, CSc., za pomoc při řešení. Dále také firmě Preciosa, a.s., za vypůjčení potřebného vybavení a všem kolegům za odborné připomínky.

# Abstrakt

Cílem této diplomové práce je seznámit čtenáře s problematikou řízení servopohonu ve vývojovém prostředí LabVIEW. Řešení probíhalo ve spolupráci se společností Preciosa, a.s.

V prvních kapitolách práce je čtenář uveden do problematiky elektrických pohonů a na těchto principech je dále rozvinuta teorie servomotorů. V dalších částech je představeno programování v prostředí LabVIEW a práce s doplňkovým modulem SoftMotion. Ve výzkumné části práce bylo účelem vytipovat servopohon dle daných požadavků a následný návrh modelových aplikací. Požadavek práce v reálném čase je zajištěn platformou CompactRIO, a tak nemusí být ošetřen dodatečnými prostředky.

Přínos této diplomové práce spočívá v rozšíření znalostí o programování v prostředí LabVIEW a především pak s knihovnou SoftMotion Module. Hlavním výsledkem práce pak je návrh modelových aplikací a diskuze jejich využití v praxi.

## **Klíčová slova:**

elektrické pohony

servopohony

LabVIEW

SoftMotion Module

modelová aplikace

# Abstract

The aim of the diploma thesis is to introduce the reader to the theory of controlling servomotors in LabVIEW development environment. The thesis was realized in collaboration with Preciosa a. s. company.

The theoretical part of the work contains introduction to electric drives and describes programming in LabVIEW. The purpose of the practical part was to choose actuators according to the given demands followed by the programming it in LabVIEW using SoftMotion Module. Design of the application was focusing on real-time work.

The benefit of this thesis lies in extension of knowledge with programming in LabVIEW, especially with SoftMotion Module library. The main result is implementation of the model application in use.

## **Keywords:**

electric drives

servomotors

LabVIEW

SoftMotion Module

model application

# Obsah

<b>1</b>	<b>ÚVOD .....</b>	<b>10</b>
<b>2</b>	<b>ELEKTRICKÉ POHONY .....</b>	<b>11</b>
2.1	SERVOPOHONY .....	13
2.1.1	<i>Porovnání servopohonů od jednotlivých společností .....</i>	<i>15</i>
<b>3</b>	<b>NATIONAL INSTRUMENTS.....</b>	<b>17</b>
3.1	LABVIEW.....	17
3.1.1	<i>Uživatelské prostředí.....</i>	<i>17</i>
3.2	SOFTMOTION MODULE.....	19
3.3	PROTOKOL ETHERCAT .....	21
<b>4</b>	<b>MODELOVÁ ÚLOHA .....</b>	<b>22</b>
4.1	SPECIFIKACE SOUSTAVY.....	22
4.1.1	<i>CompactRIO 9067 .....</i>	<i>23</i>
4.1.2	<i>AKD servoměnič .....</i>	<i>23</i>
4.1.3	<i>Servopohon AKM .....</i>	<i>24</i>
4.2	HARDWAROVÉ ZAPOJENÍ SOUSTAVY .....	25
4.3	SOFTWAREVÁ KONFIGURACE .....	27
4.4	VYTVOŘENÉ PROGRAMY .....	30
4.4.1	<i>Jednoosé řešení.....</i>	<i>30</i>
4.4.2	<i>Dvouosé řešení .....</i>	<i>33</i>
4.4.3	<i>Víceosé řešení .....</i>	<i>36</i>
<b>5</b>	<b>SHRnutí DOSAŽENÝCH VÝSLEDKŮ .....</b>	<b>38</b>
<b>6</b>	<b>ZÁVĚR.....</b>	<b>39</b>
	<b>SEZNAM LITERATURY .....</b>	<b>40</b>
	<b>PŘÍLOHY .....</b>	<b>42</b>
6.1	PŘÍLOHA A.....	42
6.2	PŘÍLOHA B.....	43



# Seznam obrázků

Obrázek 1 - Schéma elektrického pohonu .....	11
Obrázek 2 - Rovina rozdělená na kvadranty pro $\omega(M)$ .....	12
Obrázek 3 - Typy konstrukce rotoru (vlevo magnety na povrchu).....	13
Obrázek 4 - Front panel (nahore) a block diagram LabVIEW 2015 .....	18
Obrázek 5 - Palety controls (vlevo) a functions.....	19
Obrázek 6 - Možnosti bloku Property Node .....	20
Obrázek 7 - Rozdíl mezi expresní funkcí (vlevo) a základními funkcemi .....	20
Obrázek 8 - Schéma modelové soustavy .....	22
Obrázek 9 - CompactRIO 9067 [12].....	23
Obrázek 10 - AKD servozsilovač [13].....	24
Obrázek 11 - Motor AKM od firmy Kollmorgen [8].....	24
Obrázek 12 - Reálné zapojení měniče.....	26
Obrázek 13 - Foto používané soustavy .....	27
Obrázek 14 - Prostředí NI MAX a konfigurace .....	28
Obrázek 15 - Konfigurace osy v modulu SoftMotion.....	29
Obrázek 16 - Testovací panel osy v modulu SoftMotion .....	30
Obrázek 17 - Front panel jednoosé aplikace .....	31
Obrázek 18 – Blokový diagram jednoosé aplikace.....	32
Obrázek 19 - Front panel dvouosé aplikace .....	34
Obrázek 20 - Blokový diagram dvouosé aplikace .....	35
Obrázek 21 – Projekt v LabVIEW .....	36
Obrázek 22 - Robot DENSO v prostředí LabVIEW [16] .....	37
Obrázek 23 - Front panel aplikace za použití funkce Arc Move .....	42
Obrázek 24 - Blokový diagram aplikace za použití funkce Arc Move.....	43

# Seznam tabulek

Tabulka 1 - Vlastnosti feromagnetických materiálů [2] .....	14
Tabulka 2 - Srovnání pohonů jednotlivých výrobců.....	15

# 1 Úvod

Automatizace je proces, který označuje použití samočinných řídicích systémů pro řízení technologických zařízení. Zatímco mechanizace, která lidem práci usnadňuje, automatizace snižuje potřebu přítomnosti obsluhy u výrobní činnosti, proces urychluje a také snižuje náklady, popřípadě nahrazuje obsluhu v prostorech pro člověka nepřístupných. Jedním ze stavebních kamenů automatizace a mechanizace jsou elektrické pohony.

V současnosti jsou elektrické pohony pro běžný život nepostradatelné. Obsluhují nepřehledné množství strojů a zařízení, od stolní pily, až po velmi přesné automatizační roboty a manipulátory. Tento rozmach elektrických motorů můžeme připsat rozvoji výkonových polovodičových součástek a vývoji výpočetní techniky jako takové v padesátých letech minulého století. Inovace v této oblasti přispěly i ke zdokonalení řízení těchto pohonů.

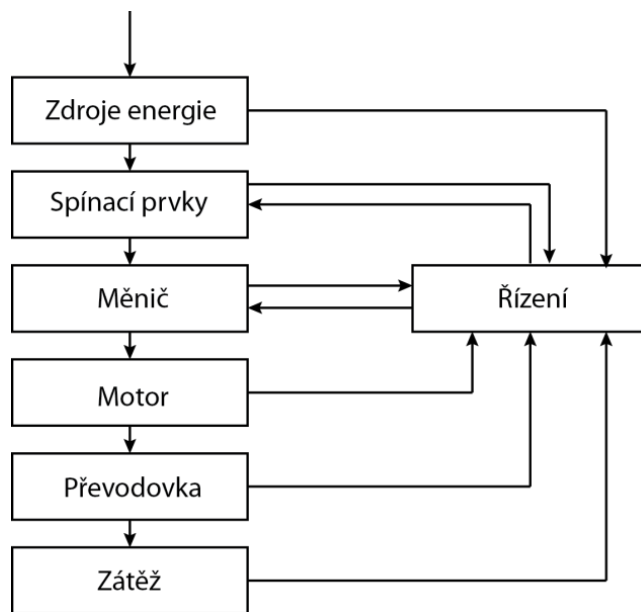
Velký rozvoj v posledních letech ale zaznamenala oblast synchronních strojů, které začaly postupně vytlačovat pohony stejnosměrné, z důvodu lepší dynamiky pohybu. Vysoká přesnost polohy a stálost otáček jsou dalšími z důvodů, proč jsou motory buzené permanentními magnety využívány v mnoha současných aplikacích. Jednou z nevýhod těchto pohonů je fakt, že jsou jejich rotory zhotovovány z materiálů vzácných zemin, a tudíž mohou být o mnoho dražší, než řešení za použití asynchronního nebo krokového motoru.

Na rozvoj v oblasti servopohonů zareagovala také americká společnost National Instruments, která započala spolupracovat s firmou Kollmorgen produkující servomotory a servozesilovače. Výsledkem této kooperace je dobrá kompatibilita servozesilovače s platformou CompactRIO a s programovým prostředím LabVIEW, ve kterém je implementována knihovna SoftMotion, poskytující nástroje pro řízení a obsluhu servopohonu. Komunikace mezi těmito heterogenními systémy je zajištěna technologií využívanou převážně v oblasti počítačových sítí, technologií ethernet, a protokolem EtherCAT, což je systém vytvořený pro automatizaci, a proto je vhodný pro real-timeové požadavky, které jsou v této technologické oblasti typické.

Tato práce si klade za cíl prozkoumat možnosti programového prostředí LabVIEW pro potřeby řízení servopohonu a následně navrhnout aplikaci využití.

## 2 Elektrické pohony

Pod pojmem elektrický pohon rozumíme soubor elektromechanických vazeb a vztahů mezi pracovními mechanismy a elektromechanickou soustavou, která má pracovní mechanismus vést předepsaným způsobem do žádaného stavu. Jako průmyslové zařízení je elektrický pohon vytvořený z patřičné kombinace přístrojů pro elektromechanickou přeměnu energie a signálů, které tuto změnu řídí. [1] Obecné schéma této soustavy viz Obrázek 1.

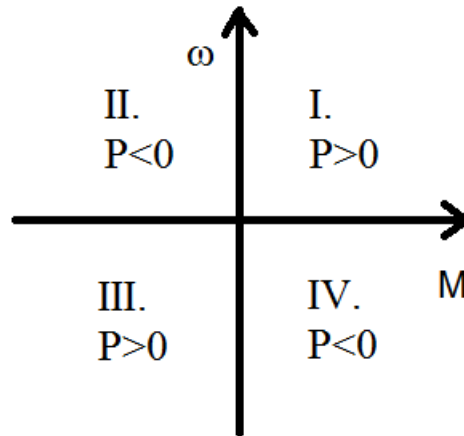


Obrázek 1 - Schéma elektrického pohonu

Za výhody elektrických pohonů lze považovat vlastnosti jako: velký pracovní rozsah ( $10^{-3}$  až  $10^8$  W), velký rozsah momentu ( $10^{-3}$  až  $10^6$  Nm) a rychlosti (10 až  $10^5$  ot/min). Elektrický pohon neprodukuje žádné výfukové zplodiny, je tichý a velmi efektivní. Nevýhodou těchto druhů pohonů je fakt, že potřebují okamžitý přívod energie ze zdroje. Poměr mezi jejich výkonem a váhou také není zanedbatelný, ve srovnání s hydraulickými mechanismy. [2]

Dělit elektrické pohony lze z mnoha hledisek. Jedním z nich je způsob pohybu na rotační a lineární. Charakter pohybu lze však měnit dodatečnými mechanismy. Z rotačního na lineární například vačkami nebo šrouby a naopak. Další dělení spočívá v průběhu pohybu rotoru. V tomto případě se jedná o spojitý (ASM,...) nebo diskrétní proces (krokové motory). Podle ovladatelnosti lze pohony dělit na jednokvadrantové, dvoukvadrantové nebo čtyřkvadrantové. Toto rozdělení vychází ze skutečnosti,

že pracovní stav motoru nebo zátěže je charakterizován momentem  $M$  a úhlovou rychlostí  $\omega$ . Z tohoto důvodu se rovina dělí na kvadranty, viz Obrázek 2. Je-li smysl momentu  $M$  shodný s úhlovou rychlostí  $\omega$ , jedná se o motor (kvadranty I., III.). V opačném případě se jedná o generátor (kvadranty II., IV.).



Obrázek 2 - Rovina rozdělená na kvadranty pro  $\omega(M)$

Dle obecné dohody se výkonový tok považuje za kladný, prochází-li elektrický výkon ze zdroje energie do pohonu a mechanický výkon, který je roven součinu  $M \cdot \omega$ , kde  $M$  je moment a  $\omega$  je úhlová rychlost, z motoru do pracovního ústrojí. Mechanický výkon motoru se pokládá za kladný, je-li smysl krouticího momentu shodný se směrem rotace. [1]

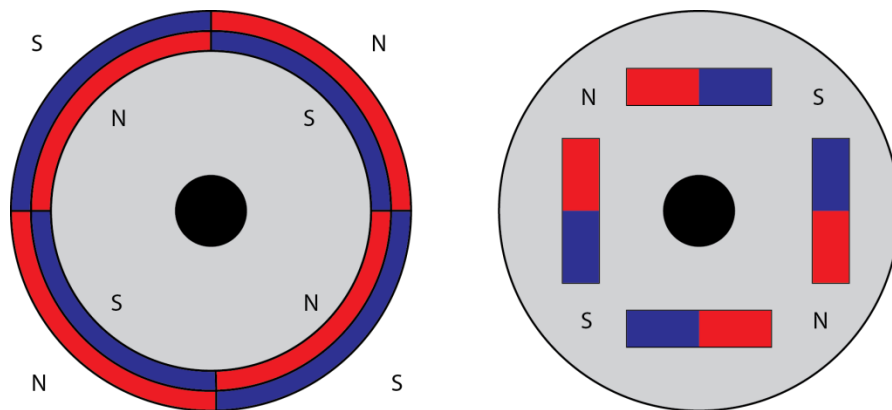
Další rozdělení el. pohonů může být z hlediska dodávané energie na střídavé, poháněné střídavým proudem, a stejnosměrné, hnané proudem stejnosměrným. Elektrické motory lze rozlišovat ještě mnoha dalšími způsoby, kterými se v této práci již nebudeme dále zabývat. Pro detailnější informace je doporučena literatura [1], [3].

V následujícím textu se již budeme bavit o elektrických pohonech nazývaných servopohony. Motory používané u servopohonů lze podle dělení z předchozích odstavců zařadit jako střídavé synchronní motory (schopné pracovat ve všech čtyřech kvadrantech) buzené permanentními magnety.

## 2.1 Servopohony

Synchronní pohony jsou stroje, ve kterých působí dvě točivá magnetická napětí, přičemž jedno je vybuzeáno vícefázovými proudy ve statoru, zatímco druhé je vybuzeáno stejnosměrným proudem v rotoru, který se otáčí synchronní rychlostí rovnou rychlosti rotace magnetického napětí statoru. Tudíž se zde nevyskytuje skluz, který vnáší jistou neurčitost a nejistotu do řízení. Odtud název synchronní.

Konstrukce synchronního stroje je taková, že v drážkách statoru je uloženo vícefázové vinutí, které může být podle konstrukce buď dvoupólové nebo vícepólové. Póly rotoru nesou buď vinutí, napájené stejnosměrným proudem přes komutátor nebo bezkontaktně rotačním transformátorem a následným usměrněním diodami na rotoru. V případě servopohonů není rotor napájen ss proudem, ale obsahuje permanentní magnety. Konstrukčně se tak velmi podobá elektronicky komutovanému motoru. Magnety mohou být buď zapuštěné, nebo umístěné na povrchu rotoru. Oba z těchto způsobů stavby rotoru mají své výhody i nevýhody. Schéma těchto dvou konstrukcí lze spatřit na Obrázku 3. [3]



Obrázek 3 - Typy konstrukce rotoru (vlevo magnety na povrchu)

Jako materiálu k výrobě permanentních magnetů se používá vzácných zemin (samarium-kobalt  $\text{SmCo}_5$ , resp.  $\text{Sm}_2\text{Co}_{17}$ , neodým-železo-bor  $\text{NdFeB}$ ) popřípadě tvrdých feritů, které jsou sice levnější, avšak také s horšími magnetickými vlastnostmi. Kvalita permanentních magnetů se porovnává podle průběhu hysterezní křivky, zejména její demagnetizační části. Charakteristické hodnoty jednotlivých materiálů jsou uvedeny v Tabulce 1. [2]

Tabulka 1 - Vlastnosti feromagnetických materiálů [2]

Složení	Rok zavedení do výroby	Remanence $B_r$ [T]	Koercivita $H_c$ [kA/m]	Energetický součin $BH_{max}$ [kJ/m <sup>3</sup> ]
Legovaná ocel	1900	0,3	20	5
AlNi	1931	0,5...0,7	32...42	7...11
AlNiCo	1940	1,2	52	40
Magneticky tvrdé ferity	1950	0,34...0,44	200...250	25...32
SmCo	1970	1...1,2	450...750	160...240
NdFeB	1983	1,2	850	220...400

Servopohony se v průmyslu zpravidla využívají tam, kde je žádoucí přesně řídit polohu nebo rychlost, nebo kde je zapotřebí velké dynamiky. Mají velkou hustotu výkonu na jednotku hmotnosti, a proto se využívají zejména v robotice. Z důvodu neustálé nutnosti sledovat polohu rotoru pro účely řízení tranzistorového měniče napájecího statorového vinutí se do servopohonů instalují snímače polohy, mezi které patří například rotační snímač (enkodér), selsyn nebo resolver. Tyto snímače lze dělit dle rozsahu poskytované polohy na absolutní, inkrementální nebo cyklicky absolutní.

Absolutní snímače udávají polohu v celém rozsahu měření otáček, tzn. ke každé pozici je přiřazena jednoznačná hodnota signálu na snímači. Mezi absolutní snímače řadíme selsyn nebo optický kotouč. Výhodou těchto čidel je, že okamžitě po zapnutí je pozice rotoru jednoznačně určena. Dalším druhem snímačů jsou inkrementální, které udávají pouze relativní polohu pohonu a musí se po zapnutí uvést do základní pozice. Výhodou je pak vysoká přesnost. Cyklicky absolutní čidla měří absolutní polohu pouze v rámci nějaké vymezené oblasti (např. jedné otáčky). [4]

Jedním z cílů této diplomové práce je vytipování servopohonu pro účely modelové aplikace. V současné době se na trhu vyskytuje velké množství společností prodávající tyto pohony, avšak mezi hlavní zástupce patří Siemens, Omron, Mitsubishi nebo Kollmorgen.

## 2.1.1 Porovnání servopohonů od jednotlivých společností

V této kapitole se zaměříme na komparaci jednotlivých typů pohonů od různých výrobců vyskytujících se v dnešní době na trhu. Toto porovnání bude probíhat na základě parametrů, kterými jsou výkon, počet otáček za minutu, přesnost enkodéru nebo kompatibilita a komunikace s ostatními periferiemi. Srovnání lze najít v Tabulce 2, kde jsou uvedeni zástupci jednotlivých firem v oblasti servopohonů. Nutno dodat, že pod pojmem servopohon se má na mysli kombinace servoměniče a servomotoru jako jednoho celku. Jednotlivé charakteristiky se mohou lišit v závislosti na dané konfiguraci jak motoru, tak servoměniče.

Tabulka 2 - Srovnání pohonů jednotlivých výrobců

	Výkon [kW]	Otáčky motoru [ot/min]	Přesnost enkodéru	Kompatibilita
Siemens Sinamics V90 + Simotics S-1FL6	0,05-7	2000-3000	$2^{13}$ - $2^{20}$	PTI, PROFINET, USS [5]
Omron R88M-K, R88D- KN[]-ECT	0,05 -15	1000-3000	$2^{17}$ - $2^{20}$	EtherCAT [6]
Mitsubishi MR-J3-xxA- EtherCAT	0,1 - 22	1000-3000	$2^{18}$	EtherCAT [7]
Kollmorgen AKD servo drive, AKM servomotor	1,1 - 32	3000-8000	$2^{20}$	EtherCAT [8]

Obecně nejsou parametry nabízených jednotek i servopohonů příliš odlišné, největší důraz tak byl kladen na schopnost komunikovat se systémem cRIO. Z tabulky je zřejmé, že společnost Siemens do svého portfolia zatím nezavedla servoměnič, který by byl schopen komunikovat přes protokol EtherCAT. Zato společnosti Omron, Mitsubishi i Kollmorgen tento protokol podporují, a tak je lze využít pro řízení platformou CompactRIO, popřípadě kartou NI PCI 7342. Aby však bylo možné



servomotor společností Omron bezproblémově řídit, musela by vzniknout dodatečná knihovna pro programové prostředí LabVIEW, jelikož společnost NI produkty od této firmy nepodporuje. Úspěšná komunikace bez dalších problémů je tak možná pouze se servomotory společností Mitsubishi a Kollmorgen. Pro oba tyto zástupce je možné do LabVIEW nainstalovat knihovnu, která by zajišťovala komunikaci přes protokol EtherCAT a zároveň by dokázala tyto motory ovládat.

Z důvodů spolupráce společností National Instruments a Kollmorgen byl pro účely této diplomové práce využit motor od druhé jmenované firmy. Stručná specifikace použitých zařízení se nachází v kapitolách 4.1.2 a 4.1.3. V následujících kapitolách se tedy výhradně zaměříme na společnost National Instruments a její produkty, popřípadě produkty spřátelené firmy, kterou je zde na mysli Kollmorgen.

## 3 National Instruments

National Instruments Corporation (někdy označována jen iniciálami NI) je americká společnost, zabývající se vývojem měřicích zařízení a strojovým viděním. Firma vznikla již v sedmdesátých letech minulého století za spolupráce Jamese Trucharda, Jeffa Kodoskyho a Billa Nowlina [9], kteří byli nespokojeni s tehdejší situací v oblasti výpočetní techniky. Kooperace těchto tří lidí dala za vznik také vývojovému prostředí LabVIEW, které tato diplomová práce využívá k řešení.

### 3.1 LabVIEW

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) je grafické vývojové prostředí, jehož základním rysem je využití nikoli textového, nýbrž grafického programovacího jazyka, někdy nazývaného také jako jazyk „G“ (neplést s G-code). Za rok vzniku LabVIEW lze považovat rok 1986 [9], kdy bylo poprvé k dispozici pro Apple Macintosh. Je běžně užíváno pro sběr a zpracování dat, ovládání různých nástrojů a průmyslovou automatizaci na mnoha platformách, mezi něž patří například Microsoft Windows, mnoho verzí UNIXu, Linux nebo OS X. Zatím poslední verze LabVIEW vyšla v srpnu roku 2015.

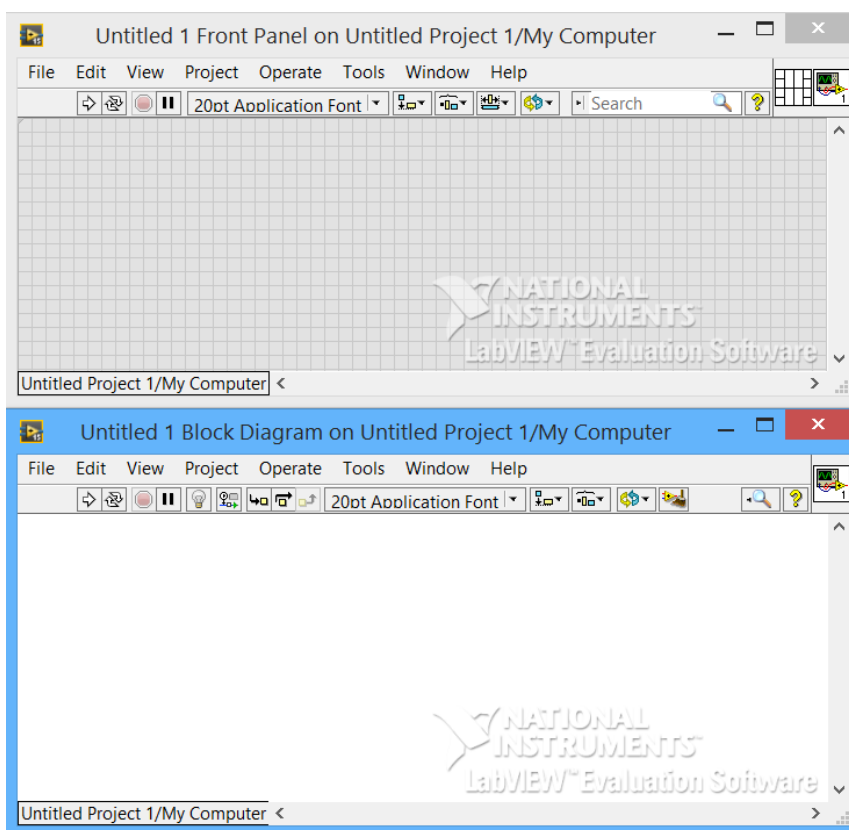
Toto vývojové prostředí využívá metodu tzv. „virtuální instrumentace“. [10] Jedná se o postup, který nahrazuje hardwarové prostředky, jež jsou finančně i časově náročnější, virtuálními nástroji, za pomoci softwaru. Tato diplomová práce byla zrealizována za pomoci LabVIEW 2015 SP1 verze 15.0.1.

#### 3.1.1 Uživatelské prostředí

Uživatelské prostředí softwaru LabVIEW lze obecně rozdělit do dvou hlavních oken: front panel, v češtině nazýván také čelní panel, a blokový diagram (block diagram). V průběhu této práce se můžeme setkat jak s těmi českými, tak anglickými výrazy, popisujícími tyto dva segmenty.

## **Front panel a block diagram**

V těchto částech lze vytvářet různé stránky programu. Jak již možná trochu napovídá název, v blokovém diagramu se vytváří základ každé aplikace, tedy zdrojový kód, obsahující jednotlivé funkce, smyčky a logické operace. Ve front panelu se vytváří rozhraní mezi uživatelem a aplikací. Obsahuje zpravidla ovládací a informační prvky. Lze sem zařadit tlačítka, číselné i textové vstupy, indikátory, grafické výstupy atp. Obě tyto části programu lze vidět na Obrázku 4.

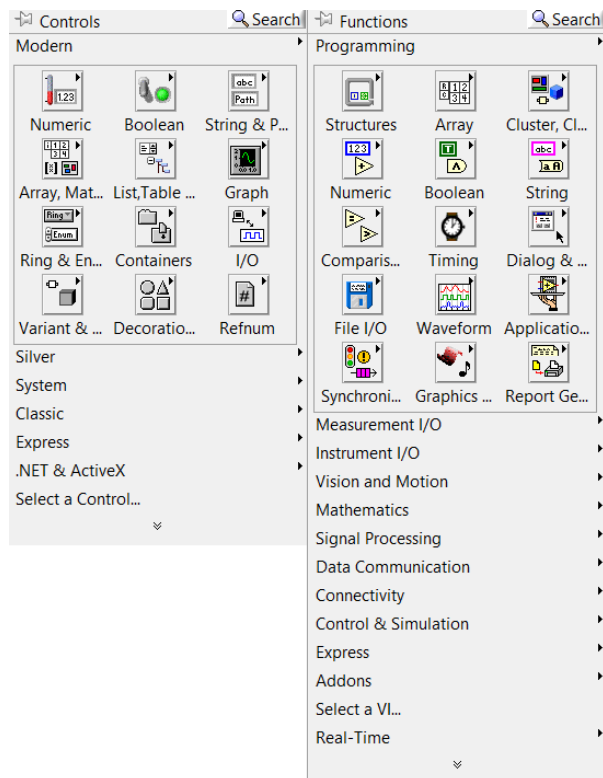


Obrázek 4 - Front panel (nahore) a block diagram LabVIEW 2015

## **Paleta Controls a Functions**

Tyto palety příslušejí jednotlivým částem programu. Na čelním panelu můžeme nalézt paletu controls, která obsahuje ovládací a indikační prvky, zatímco v okně blokového diagramu můžeme vyvolat paletu functions. Zde nalezneme funkční prvky programu, kterými jsou například smyčky, časování, logické a numerické operace, atp.

Každá z těchto palet je podle použití přehledně rozdělena do mnoha dalších subpalet. Nabídka functions tak obsahuje kategorie například pro měření, zpracování signálu, zpracování obrazu atd. Ukázky obou palet představuje Obrázek 5.

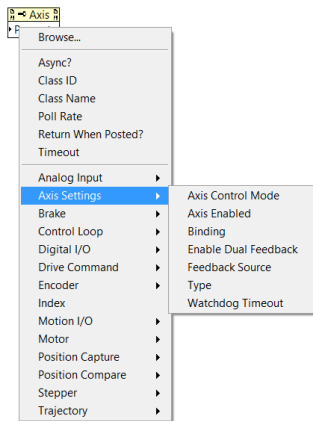


Obrázek 5 - Palety controls (vlevo) a functions

## 3.2 *SoftMotion Module*

Tato nadstavbová knihovna přináší do grafického prostředí LabVIEW možnost, jak konfigurovat, testovat a řídit synchronní servopohony. Pro účely této diplomové práce byla využita verze 15.5.0.

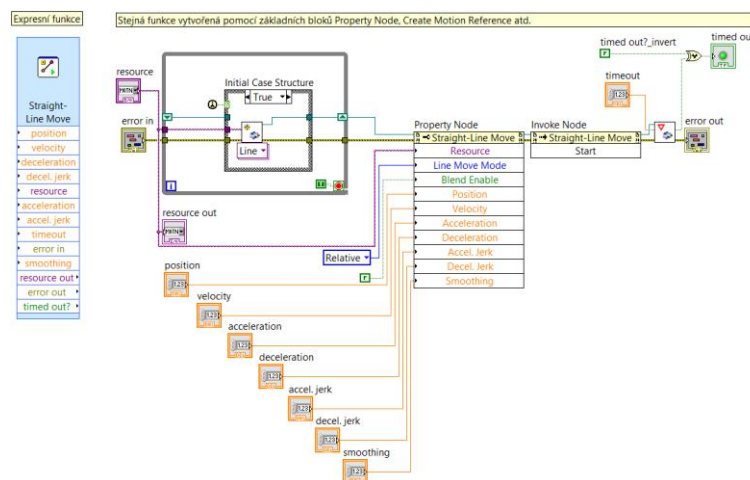
Po instalaci lze tento modul nalézt v paletě funkcí v podkategorii Vision and Motion. Obsahuje dva typy použitelných funkcí. Jedním z těchto typů jsou základní bloky Property Node, Invoke Node, Create Motion Reference, Destroy Motion Reference a Wait Until Done. Za pomoci těchto pěti funkcí lze zhotovit mnoho druhů pohybu, informačních ukazatelů atp., avšak pro běžného uživatele je tento typ programování poněkud nepohodlný. Vyplývá to z faktu, že samotné nastavení těchto funkcí je velmi zdlouhavé, neboť rozsah poskytovaných možností je velmi široký. Pro ukázkou nastavení uzlu Property Node viz Obrázek 6.



Obrázek 6 - Možnosti bloku Property Node

Skutečnost, že třídu typu Motion lze navíc změnit z Resources (zdroje), kde se spolu s osou vyskytují ještě souřadnice a tabulka, na Interface (rozhraní), dále rozšiřují možnosti nastavení bloku Property Node téměř dvojnásobně.

Z výše zmíněných důvodů proto lze v modulu SoftMotion nalézt předprogramované bloky v podobě expresních funkcí, které představují druhý typ funkcí. Tyto předpřipravené kusy kódu usnadňují a urychlují práci s knihovnou, neboť obsahují základní typy chodu motoru a konfigurační, případně informační prvky. Vytváříme-li však aplikaci pro atypickou nebo konkrétní úlohu, nemusí nám expresní funkce stačit, a proto jsme nuceni sáhnout po nižších programovatelných prvcích v podobě Property Node atp. Rozdíl mezi těmito dvěma způsoby programování lze spatřit na Obrázku 7. Vlevo lze vidět jeden blok expresní funkce, který zaujímá stejnou funkci jako množina prvků a struktur vpravo, využívající základní funkce.



Obrázek 7 - Rozdíl mezi expresní funkcí (vlevo) a základními funkcemi

V modulu SoftMotion jsou také tři základní typy „Motion resources“, tedy zdroje pohybu, které lze dělit na:

- „Axis“ (osa), kterou lze libovolně nakonfigurovat v závislosti na použitém hardwaru.
- „Coordinate space“ (souřadnicový systém), který slouží pro svázání více os do jednoho celku. Další výhodou je také fakt, že tyto osy jsou automaticky sesynchronizovány, aby svůj pohyb začínaly i končily ve stejný okamžik.
- „Table“ (tabulka), která se zde dělí na dva druhy. Jedním z nich je „Contour table“, do které se zaznamenávají body, které má motor následovat zvoleným typem spline křivky. Druhým typem je „Camming table“, která slouží pro zaznamenání dat pro elektronickou vačku. Tyto tabulky lze v obou případech uchovávat v podobě textového souboru, nebo v případě „contour table“ ji lze nahrát do zásobníku.

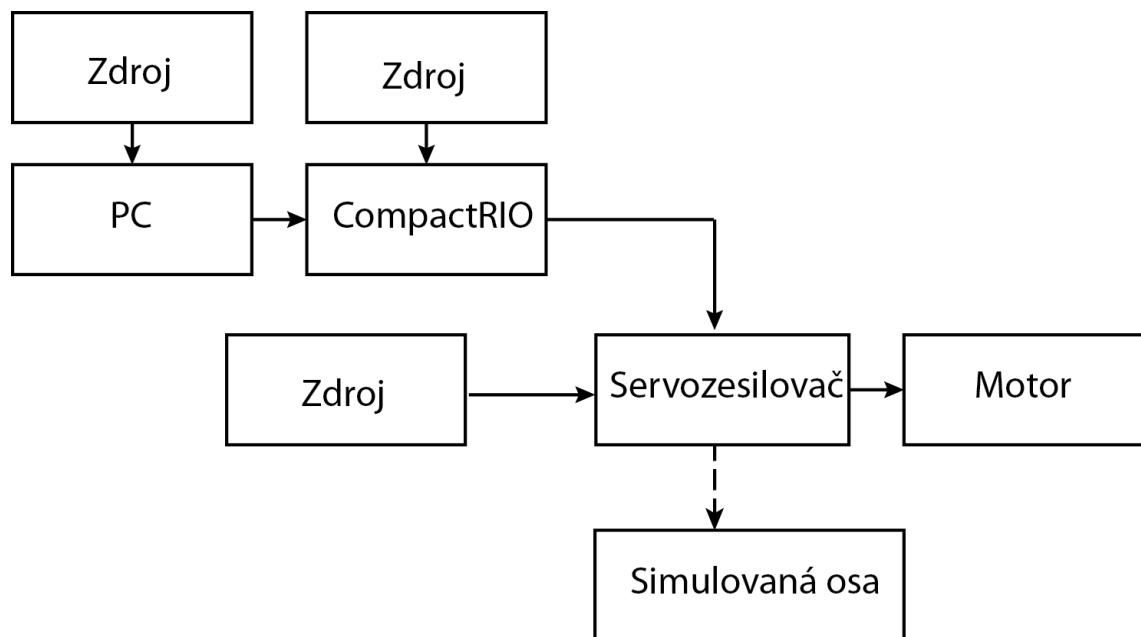
### **3.3 Protokol EtherCAT**

EtherCAT byl světu představen v roce 2003 společností Beckhoff Automation. Požadavky na tento protokol byly rychlost, přesná synchronizace a přijatelná cena hardwaru. Základy jsou postaveny na již dobře známé a osvědčené technologii ethernet.

Funkce protokolu EtherCAT je založena na preposílání datového rámce z hlavního kontroléru nazývaným „master“ (do češtiny lze volně přeložit jako „pán“) do dalších podřadně zapojených zařízení v soustavě („slaves“ = otroci). Tyto „slave“ zařízení čtou a zapisují data za chodu. Tento typ zpracování dat je v cizích jazycích nazýván jako „on-the-fly“, tedy „za letu“. Schopnost takto číst data je umožněna díky technologii ESC (EtherCAT Slave Controller), která je realizována jako integrovaný obvod ASIC nebo programovatelné pole FPGA. Tato vlastnost tak nevnáší do systému žádné větší zpoždění datového toku. Vysílaný rámec tímto způsobem putuje až ke konečnému zařízení, načež ho posílá stejnou cestou zpátky. Základním předpokladem této komunikace je fakt, že pouze „master“ může vysílat rámec, obsahující potřebné informace, a ostatní „slave“ zařízení mohou zprávu pouze upravovat. [11] Pro další zdroje informací ohledně EtherCAT a jeho vlastností můžete navštívit webové stránky společnosti Beckhoff Automation na adrese [www.ethercat.org](http://www.ethercat.org).

## 4 Modelová úloha

Cílem této diplomové práce bylo prozkoumat možnosti programového prostředí LabVIEW pro potřeby řízení synchronního motoru buzeného permanentními magnety. Pro tento účel vznikla modelová úloha sestávající ze dvou synchronních os, průmyslového počítače CompactRIO (cRIO) od společnosti NI a osobního počítače. Komunikace počítače s CompactRIO a CompactRIO se servoměničem je vedena přes ethernetový kabel. Obrázek 4 znázorňuje obecné schéma soustavy.



Obrázek 8 - Schéma modelové soustavy

CompactRIO v této struktuře zajišťuje možnost práce v reálném čase pro celou aplikaci, jelikož operační systém Microsoft Windows běžícím na osobním počítači není schopen tyto požadavky splnit (operační systém Windows je časově nedeterministický). Celý program vytvořený na osobním počítači se tak musí importovat do cRIO, které následně vykonává celou operaci.

### 4.1 Specifikace soustavy

V následujících kapitolách se zaměříme na stručnou specifikaci jednotlivých komponent modelové úlohy. Jak již nastínilo obecné schéma z Obrázku 4, byl použit následující hardware:

- 1x NI CompactRIO 9067
- 1x Kollmorgen AKD servo-měníč
- 1x Kollmorgen AKM servomotor
- Osobní počítač s MS Windows 8.1 a LabVIEW 2015 + Real-Time Module + SoftMotion Module

### 4.1.1 CompactRIO 9067

Kontrolér CompactRIO obsahuje kombinaci dvoujádrového procesoru, FPGA pole a osmi slotů pro umístění C modulů. V soustavě tak nahrazuje osobní počítač, který není schopen pracovat v reálném čase („real-timeově“). Je to tak ideální řešení v průmyslových aplikacích vyžadující řízení technologie, sběr dat a jejich zpracování.

Charakteristika CompactRIO:

- Dvoujádrový procesor ARM Cortex-A9 na frekvenci 667 MHz a FPGA obvod Xilinx Artix-7
- 512 MB DDR3 operační paměť, 1 GB paměti pro záznam hodnot
- Operační systém NI Linux Real-Time
- Dva sloty pro Gigabit Ethernet, jeden USB slot, jeden sériový port [12]

Pro ukázkou platformy CompactRIO viz Obrázek 9.



Obrázek 9 - CompactRIO 9067 [12]

### 4.1.2 AKD servoměníč

Pro účely této práce byl použit servoměníč od firmy Kollmorgen s označením AKD-P00606-NAEC-0000. Výhodou měničů tohoto typu je široká nabídka způsobu komunikace a snadné připojení k různým řídicím systémům. Schopnost pracovat s protokolem EtherCAT je další nespornou výhodou pro implementaci tohoto měniče,



zvláště v kombinaci s platformou CompactRIO. Jmenovitý proud tohoto měniče činí 6 A, zatímco špičkový proud může dosáhnout hodnoty 18 A a výstupní výkon 2000 W. [13] Servoměnič AKD lze spatřit na Obrázku 10.



Obrázek 10 - AKD servozsilovač [13]

### 4.1.3 Servopohon AKM

V rámci této diplomové práce byl využit motor AKM31E-ANCNC-00, což je bezkartáčový servomotor s rotorem ze slitiny neodym-železo-bor. Výhodou bezkartáčových motorů je fakt, že se zde nevyskytuje žádný komutátor, a tudíž žádné kartáče, které se mohou opotřebovat a ve kterých mohou zde vznikat ztráty. Maximální rychlost činí 8000 otáček za minutu a jmenovitá rychlost 6000 otáček při přívodu napětí 240 V. Další charakteristiky lze najít v technické dokumentaci zařízení [14]. Pro ukázkou použitého motoru viz Obrázek 11.



Obrázek 11 - Motor AKM od firmy Kollmorgen [8]

## **4.2 Hardwarové zapojení soustavy**

V následující kapitole bude stručně popsáno, jak byl systém zapojen. Pro uvedení celé soustavy do provozu byly potřeba dva usměrňovače, neboť jak cRIO, tak servozesilovač AKD požadují stejnosměrné napájení 24 V. [15]

Dalším bodem v sestavování bylo spojení jednotlivých zařízení ethernetovým kabelem, který zajišťuje tok dat mezi konkrétními částmi. Zapojení motoru do systému bylo zrealizováno pomocí příslušných kabelů pro přenos elektrické síly a informace o pozici z enkodéru.

Posledním krokem pro úspěšné zapojení soustavy bylo přivedení střídavého napětí o hodnotě 230 V na vstup servoměniče a zapojení bezpečnostních prvků, zamezujícím pohyb servomotoru. Mezi tyto části patřilo zejména přivedení kladného pólu stejnosměrného napájení na STO (Safe Torque Off) vstup servozesilovače a stejný postup byl zvolen pro signály enable a DCOM na digitálních vstupech zařízení.

STO funkcionalita zajišťuje, že v motoru se nebude vyskytovat žádná momentotvorná energie, která by mohla přivést motor do aktivního stavu, a chrání před neúmyslným spuštěním motoru. Tento mechanismus také zabraňuje před neočekávaným restartem systému a vychází z normy EN 60204-1, sekce 5.4. Z bezpečnostních důvodů tak bývá tento vstup připojen na světelnou závoru nebo podobné bezpečnostní prvky, zamezující pohybu člověka v blízkosti běžícího motoru, popřípadě je využit pro tlačítko nouzového vypnutí systému. Kompletní zapojení servoměniče lze spatřit na Obrázku 12.



Obrázek 12 - Reálné zapojení měniče

Na Obrázku lze vidět, že vstupy s označením X1 označují logický vstup pro 24 V DC a STO. Vstupy X8 na pozici 3 a 4 jsou signály enable a DCOM, které jsou také přivedeny na zdroj stejnosměrného napětí. Výstup z portu označeného X2 je napájení pro synchronní motor a vstup X3 je střídavé napájení servoměniče. Komunikace mezi CompactRIO a servoměničem zajišťuje port s označením X5. Zpětná vazba z enkodéru umístěného v motoru je vedena přes port X10.

Obrázek 13 představuje kompletní zapojení mechanismu. Pro zapojení druhé osy nebyla k dispozici potřebná kabeláž, a tak vše probíhalo pouze prostřednictvím jedné reálné osy, zatímco další osy byly simulované v prostředí LabVIEW.



Obrázek 13 - Foto používané soustavy

## 4.3 Softwarová konfigurace

Softwarová konfigurace byla v celkovém řešení časově nejnáročnější. Muselo být nainstalováno mnoho aplikací jak do osobního počítače, tak do platformy CompactRIO. Mnohá úskalí, která budou okomentována v následujících odstavcích, přinesla také samotná konfigurace soustavy.

Prvním krokem řešení bylo nainstalování příslušné verze LabVIEW spolu s ovladači pro cRIO verze 15.5 a nadstavbovými knihovnamy Real-Time Module a Softmotion Module, které zajišťují kompatibilitu a komunikaci mezi jednotlivými systémy.

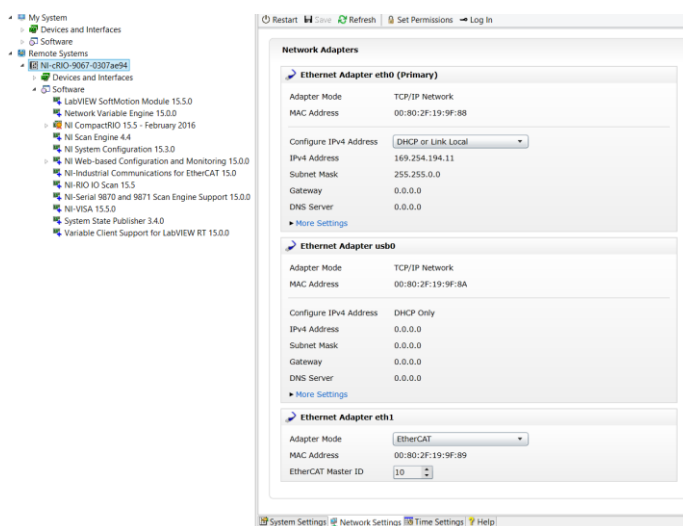
Následujícím krokem bylo nainstalování operačního systému do kontroléru cRIO. V tomto případě byla zvolena nejnovější verze systému, verze 15.5. Další podmínkou funkcionality bylo nainstalování NI Scan Engine a modulu SoftMotion, stejně jako u verze pro PC. Pro instalaci a kontrolu softwaru byl využit program NI MAX, pomocí kterého lze konfigurovat a spravovat jak hardwarové, tak softwarové prostředky.

Dále musí být upraveno nastavení síťové karty v PC tak, aby byla schopna s kontrolérem komunikovat. Toto nastavení spočívalo v zadání pevné IP adresy v konfiguraci protokolu TCP/IPv4. Adresu, kterou je potřeba zadat, lze zjistit také

z programu NI MAX. V našem případě mělo cRIO adresu 169.254.194.11 a masku podsítě 255.255.0.0.

Aby bylo CompactRIO schopno komunikovat se servozsilovačem, musí být také řádně nastaveno. Po připojení periférií v podobě servozsilovačů se v záložce „Network Settings“ musí přepnout mód adaptéru příslušného portu na volbu „EtherCAT“ a nastavit příslušnou adresu, pod kterou se bude „master“ zařízení hlásit. V rámci této diplomové práce bylo zvoleno ID hodnoty 10 pro lepší orientaci a zapamatovatelnost.

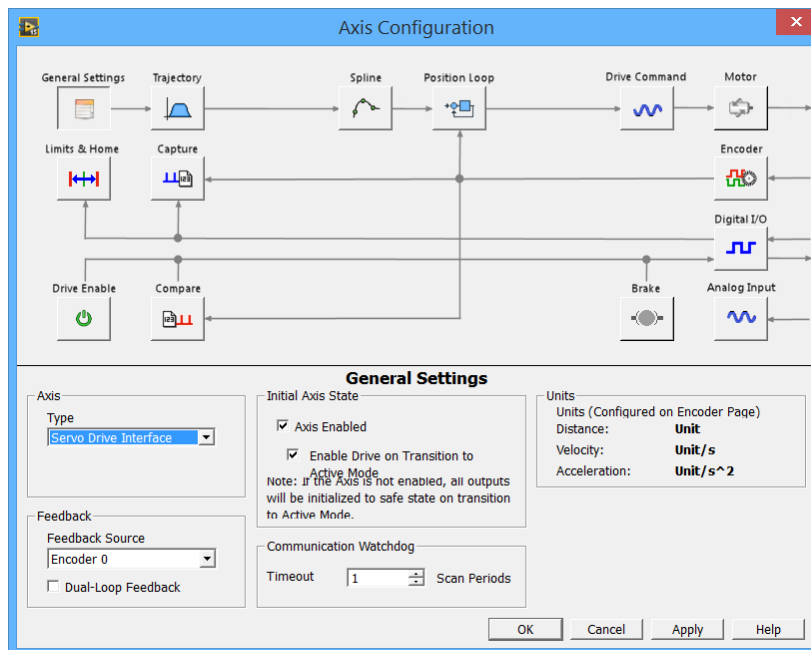
Finální nastavení cRIO a jeho operačního systému lze spatřit na Obrázku 14. Vlevo v podkategorii „Software“ lze vyčíst jednotlivé programové vybavení kontroléru 9067, zatímco v pravé části ilustrace se nachází ukázka jeho správného nastavení v rámci síťových adaptérů.



Obrázek 14 - Prostředí NI MAX a konfigurace

Další konfigurace probíhala již v rámci LabVIEW. Po předešlé konfiguraci bychom měli být schopni přidat kontrolér cRIO do nově založeného projektu pomocí možnosti Tools > Real-Time Module > Project Wizard. V případě, že se CompactRIO v tomto průvodci nezobrazuje, byla udělána chyba v předešlých krocích, nebo není nainstalovaný potřebný software v platformě cRIO.

V modulu SoftMotion existuje mnoho možností a nástrojů jak servoosy konfigurovat a testovat. Nejdříve však musí být osa přidána do projektu pomocí kontextové nabídky New > SoftMotion Axis. V tomto stavu je již možno osu konfigurovat vyvoláním nabídky stiskem pravého tlačítka a zvolení položky „Properties“. Tímto postupem bychom se měli dostat do nabídky z Obrázku 15.



Obrázek 15 - Konfigurace osy v modulu SoftMotion

V této konfiguraci musíme zajistit, aby byly řádně nastaveny hodnoty pro enkodér a limity dopředného, resp. reverzního chodu motoru. Z přiložené dokumentace lze zjistit, že správná hodnota pulsů na otáčku pro enkodér činí 1048576, tedy  $2^{20}$ , jak bylo zmíněno v předchozích kapitolách. Nabídka konfigurace také umožňuje změnu spline křivky podle požadavků z kubické B křivky na Catmull-Rom. Dalšími parametry, které lze v této nabídce upravovat, jsou hodnoty proporcionálního, integračního a derivačního zesílení v záložce polohového řízení.

Pro základní odzkoušení správnosti konfigurace osy je dále v modulu SoftMotion implementován testovací panel, kde lze s motorem provádět základní operace, kterými jsou například nabuzení a vypnutí osy, pohyb o určitý počet otáček, popřípadě úhel. Tento panel lze spatřit na Obrázku 16. Vyskytuje se zde také ukazatel aktuální pozice, rychlosti, přítomny jsou kontrolky informující obsluhu o stavu pohonu, popřípadě o nastavení či probíhajícímu pohybu. Tento způsob řízení však není dostačující pro náročnější aplikace a mechanismy, a proto je potřeba vytvořit komplexnější programy, které budou těmto požadavkům vyhovovat.



Obrázek 16 - Testovací panel osy v modulu SoftMotion

## 4.4 Vytvořené programy

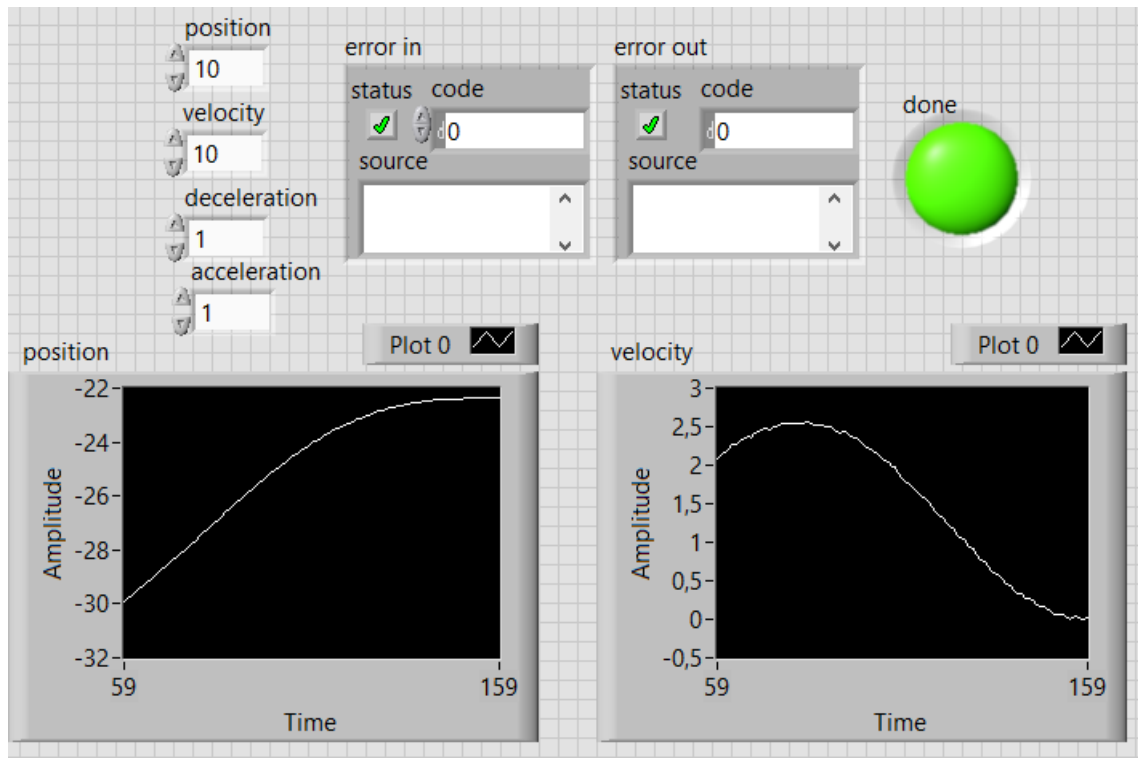
V následujících částech práce se zaměříme na konkrétní aplikace vytvořené pomocí vývojového prostředí LabVIEW. Kapitoly jsou vzestupně seřazeny podle náročnosti na hardware od jednoosého mechanismu až po teoretické využití robotického manipulátoru v nejnáročnějších technologických mechanismech.

### 4.4.1 Jednoosé řešení

Základem této aplikace bylo porozumění programování servopohonu v prostředí National Instruments. Z důvodu názornosti a nenáročnosti bylo zvoleno ovládání pomocí expresních funkcí, které nám nahrazují náročnější postupy v případě funkcí Property Node a Invoke Node.

Na Obrázku 17 se vyskytuje ukázka čelního panelu vytvořené aplikace. Obsahuje konfigurační prvky v podobě posuvných ovladačů, kterými lze ovládat cílovou pozici motoru, rychlost pohybu, rychlost akcelerace a decelerace. Dále jsou zde uvedeny indikátory chyb, kde se vypisují chybová hlášení, ukončil-li se program

neočekávaným způsobem, popřípadě byla-li vytvořena chyba ve zdrojovém kódu aplikace. Zelená kontrolka nám indikuje, proběhnul-li program bez závad. Dva vyskytující se grafy podávají informaci o aktuální poloze a rychlosti jedné osy mechanismu. Jednotky jsou udávány v počtu otáček v případě ukazatele pozice, respektive počtu otáček za sekundu v grafu rychlosti.



Obrázek 17 - Front panel jednoosé aplikace

Tento panel však není jedinou součástí aplikace. Dalším nezbytným segmentem je samozřejmě zdrojový kód, který je uveden na Obrázku 18. Kód obsahuje celkem čtyři expresní funkce, které nám umožňují ovládání motoru. Jsou to funkce: Power, Straight-Line Move, Read a Stop Move. Funkce Power nám povolí s motorem manipulovat a nabudí ho, proto musí být na začátku každé aplikace. Ještě před tímto segmentem se však vyskytuje nulování chyb a vypnutí kontrolní LED diody na front panelu, což je zajištěno strukturou „Flat Sequence“, která je vykonána bezprostředně po zapnutí programu.

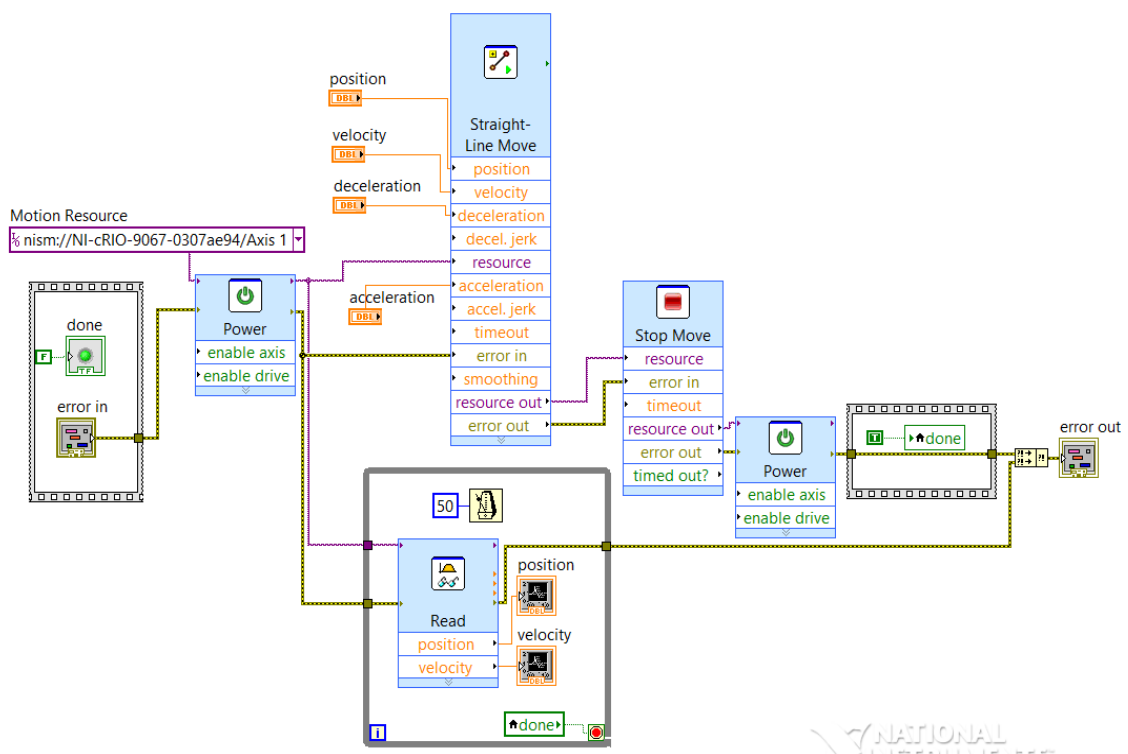
Aby bylo bezprostředně určeno, který prvek má být spuštěn, vstupuje do bloku „Power“ zdroj typu „Axis 1“, který máme k dispozici v kořenovém adresáři projektu a je připojen na kontrolér cRIO. To lze dokázat tím, že před názvem osy je uvedena cesta, kde se zdroj nachází. Dále se program dělí na dvě nezávislé cesty.



První část vykonává samotný pohyb. Nutností je v první řadě zadat požadované hodnoty na vstupy „position“, „velocity“, „deceleration“ a „acceleration“ bloku „Straight-Line Move“. Tento samotný segment zaručuje pohyb motoru a navazujícími funkčními bloky „Stop Move“ a „Power“ je jen zajištěno bezpečné zastavení pohybu a vypnutí příslušné osy.

Druhou cestou v programu je čtení poskytovaných informací ze zdroje. K tomuto účelu se zde vyskytuje blok „Read“. Tato funkce musí být umístěna ve „while“ smyčce, neboť chceme kontinuální čtení dat po dobu chodu motoru. Funkce má mnoho parametrů, avšak v této aplikaci byly vybrány pouze parametry udávající informaci o aktuální pozici a rychlosti.

Tento jednoduchý příklad nastiňuje, jak je servopohon v prostředí LabVIEW řízen. Vytvořená aplikace by mohla být použita v jednoduchých procesech, ve kterých není zapotřebí složitějších algoritmů a dynamiky. Mohla by tak být například využita při realizování pohonu pro podavače popřípadě pásového dopravníku.



Obrázek 18 – Blokový diagram jednoosé aplikace

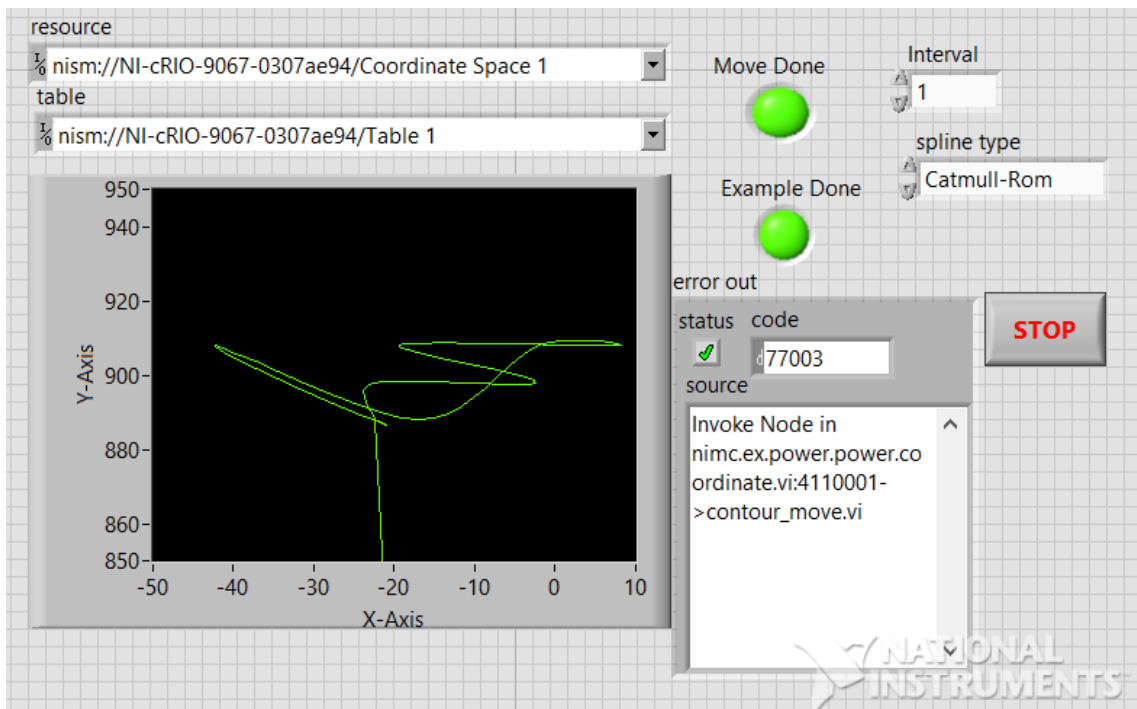
## 4.4.2 Dvousé řešení

Tato aplikace ukazuje, jak lze naprogramovat dvě na sobě závislé servoosy. Jelikož byla fyzicky k dispozici pouze jediná, druhá musela být simulována virtuálně, což modul SoftMotion dovoluje. Do řídicího projektu tak musí být vložen další prostředek pohybu v podobě „coordinate space“, který musí být oběma osami definován. Chceme-li, aby osy procházely určitými body v prostoru, musíme do projektu dodefinovat i „contour table“, kterou nastavíme pro dvě osy o určitém počtu požadovaných bodů.

Na níže uvedeném Obrázku 19 můžeme vidět čelní panel druhé aplikace, zaměřené na svázání dvou os do jednoho souřadnicového systému. Vyskytují se zde dvě rolovací menu, kde lze vybrat potřebné prostředky pohybu, popřípadě tabulku. Dále se zde vyskytuje graf, kde je zobrazen průběh sledované křivky. Nutno doplnit, že na ose X se nacházejí údaje z fyzické osy, zatímco Y osa uvádí data z nasimulované osy.

Dvě kontrolní LED diody nás informují o dokončeném pohybu, respektive dokončeném programu. Vstupní hodnota „Interval“ určuje, jaký má být časový interval mezi jednotlivými zadanými body, jelikož v nabídce pro pohyb po kontuře není možnost, jak definovat rychlost, popřípadě zrychlení pohybu daných os. Dalším ovládacím prvkem na front panelu je nouzové tlačítko „STOP“, které v případě nesrovnalostí, popřípadě chyby v programu celý program ukončí.

Možností, jak průběh křivky ovlivnit, spočívá v nastavení typu „spline type“, kde máme na výběr z „Catmull-Rom“ typu nebo „Cubic B spline“. Tyto možnosti si nejsou rovny, jelikož z definice těchto křivek vyplývá, že zatímco „spline“ typu „Catmull-Rom“ prochází přesně zadanými body, druhý typ prochází pouze počátečním a koncovým bodem, zatímco k ostatním bodům se pouze přiblíží. Objevuje se zde také nezbytný blok pro hlášení chyb, který nesmí chybět v žádné reálné aplikaci.

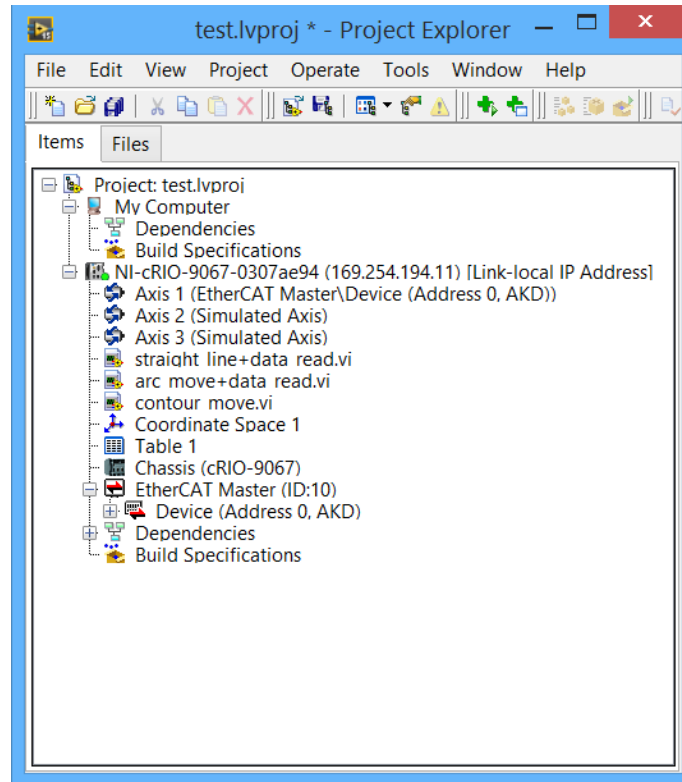


Obrázek 19 - Front panel dvouosé aplikace

Obrázek 20 ilustruje, jak vypadá blokový diagram dvouosé aplikace. Lze spatřit, že obecná struktura programu se od předchozího příkladu liší pouze minimálně. Prvním krokem je opět nulování chyb a vypnutí kontrolních indikátorů přes smyčku „Flat Sequence“. Následuje inicializace „Motion resource“, avšak v tomto případě se jedná již o prostředek typu „Coordinate Space“, který současně nabudí obě přiřazené osy. Dále se zde schéma opět rozděluje, tentokrát na tři různé segmenty. Vrchní část programu zajišťuje chod pohonu. Vyskytují se zde dva bloky funkce „Contour Move“, avšak každý s jiným parametrem. První zajišťuje vložení zadané tabulky do zásobníku, popřípadě její aktualizaci, běží-li program kontinuálně. Dále do této funkce vstupuje datový typ Boolean konstantně nastaven na hodnotu true, který zajišťuje funkci přepsání hodnot z tabulky. Na první blok dále navazuje funkce, která poskytuje samotný pohyb motoru. Vstupními parametry jsou zadané body, uložené v tabulce v podobě lokální proměnné, typ „spline“ křivky a interval. Dále se ve schématu vyskytuje odbuzení motorů a následné přepnutí lokální proměnné s názvem „Move Done“, která zajišťuje jak signalizaci na čelním panelu, tak ovládá obě while smyčky, které má schopnost ukončit.



Obrázek 21 znázorňuje, jak může vypadat konečný projekt. Vidíme, že se zde vyskytují všechny výše popisované programy, fyzické i simulované osy, souřadnicový systém, tabulka i připojený kontrolér CompactRIO spolu s „master“ zařízením v podobě servozesilovače.



Obrázek 21 – Projekt v LabVIEW

### 4.4.3 Víceosé řešení

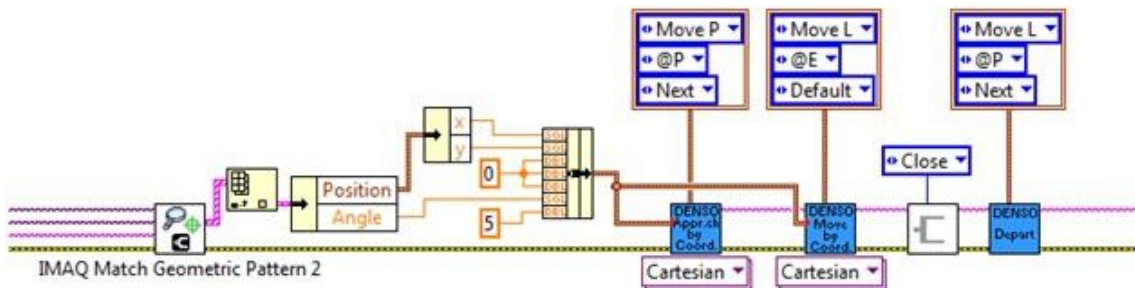
Jak již bylo naznačeno na Obrázku 21, v projektu lze přidat a manipulovat i s více než dvěma programovatelnými osami. Obtíže však nastávají, mají-li se pohybovat po zadané trajektorii, jako tomu bylo například u expresní funkce „Contour Move“. K řízení více os jsou v modulu SoftMotion k dispozici pouze funkce „Arc Move“ nebo „Helical Move“. Obě tyto funkce však nedokáží projet zadanými body, nýbrž opisují kruhovou trajektorii.

Aby mohly být osy řízeny ve třech osách současně a zároveň aby dokázaly sledovat požadovanou křivku, musel by být ručně vytvořen nový blok expresní funkce. V této funkci by musel být algoritmus na stanovení trajektorie ze zadaných bodů, dále

by musela být vyřešena synchronizace motorů a v neposlední řadě také 3D graf, který by usnadňoval orientaci na 2D zobrazovacím zařízení v podobě monitoru.

Další možností, jak řídit víc než dvě osy v modulu SoftMotion, je použití bloků „Camming“, popřípadě „Gearing“. Tyto funkce slouží pro vačkové, respektive převodové mechanismy, avšak tato práce se jimi dále nezabývala.

Alternativním řešením pro požadavky víceosých mechanismů by mohlo být za využití víceosého robotu. Ten má již ve své řídicí jednotce vše nastaveno, a tak nezbyvá nic jiného, než mu zadat požadovanou trajektorii pohybu. Práce s automatizačním robotem nebývá zpravidla nijak složitá, avšak problém by mohl nastat, chtěli-li bychom tento mechanismus implementovat do vývojového prostředí LabVIEW. Pro tyto účely však existují nadstavbové knihovny. Může takto být řízen například robot známé společnosti KUKA Roboter, dále také robot firmy DENSO Robotics nebo například robot Mitsubishi. Obrázek 22 představuje ukázkou programování robota DENSO.



Obrázek 22 - Robot DENSO v prostředí LabVIEW [16]

## 5 Shrnutí dosažených výsledků

Řešení diplomové práce probíhalo bez větších problémů, avšak určité těžkosti se během tvoření díla vyskytly. Především bylo třeba prostudovat mnoho materiálů ohledně protokolu EtherCAT, kontroléru CompactRIO, ještě před samotným začátkem návrhu řešení. Dále bylo třeba provést instalace potřebného softwaru a operačního systému do periférií a v neposlední řadě nastavit komunikace mezi jednotlivými zařízeními.

Samotné programování aplikací bylo již vyvrcholením celé práce. K tomuto účelu byly k dispozici ukázkové příklady ve vývojovém prostředí LabVIEW. Vytvořená VI mohou čtenáři naznačit, jak se dají řídit servopohony za pomoci modulu SoftMotion, i přesto, že v práci nebyly představeny všechny expresní funkce z nabídky knihovny. Obecně se však již způsoby použití dalších bloků nijak výrazně neliší od zde prezentovaných. Například použití funkce „Arc Move“ můžete dále nalézt v přílohách A a B.

Příklad dvouosého mechanismu, který byl uveden v této diplomové práci v kapitole 4.4.2, by mohl být dále využit pro řešení pohybu v souřadnicovém systému XY. Systém by však musel být doplněn o potřebné mechanické vazby a zároveň by musely být správně nakonfigurované osy v rámci mechanismu. Soustava by dále mohla být osazena kamerovým systémem, který by snímal umístěné předměty na zadaných souřadnicích a zároveň vyhodnocoval jejich parametry.

## 6 Závěr

V této diplomové práci bylo představeno, jak lze konfigurovat a řídit synchronní servopohony buzené permanentními magnety ve vývojovém prostředí LabVIEW. Práce je dělena do dvou částí. První z nich je segment kapitol, zabývající se rešerší zadané problematiky. Výzkumná část díla je zaměřena na vývoj modelové aplikace, kde jsou ukázány možná řešení zadaného problému.

Z důvodů zmíněných v kapitole 2.1.1 byly pro realizaci modelové úlohy využity servopohony z nabídky společnosti National Instruments a platforma CompactRIO taktéž od NI. CompactRIO zaručuje real-timeový chod programu, a tudíž ho není potřeba ošetřovat dalšími prostředky.

Komunikace mezi jednotlivými součástmi soustavy probíhala přes ethernetový kabel s pomocí protokolu EtherCAT. Pro programování jednotlivých aplikací bylo využito nadstavbové knihovny SoftMotion, která poskytuje všechny potřebné prostředky k úspěšnému vytvoření řídicího programu.

Pro konfiguraci synchronních os bylo využito nástroje z modulu SoftMotion, ve kterém lze měnit velké množství parametrů. Důležitým segmentem v nastavení je seřízení zpětné vazby, v našem případě enkodéru, kde se musí nastavit správné parametry (jednotky, počet pulsů na otáčku, atd.) pro spolehlivý chod aplikace. Jakmile byly osy seřizeny, bylo možné začít s vývojem programů.

Zdrojový kód aplikace bylo možno vytvořit několika možnými způsoby. Knihovna SoftMotion má v nabídce buď množinu expresních funkcí, nebo skupinu bloků, které lze využít pro detailnější a pro některé případy výhodnější řízení. Tuto nižší strukturu však při návrhu nebylo nutné využít, poněvadž expresní funkce poskytovaly dostatečné možnosti nastavení.

Aplikace prezentované v kapitolách 4.4.1 a 4.4.2 dostatečně představují vytváření řídicích algoritmů v prostředí LabVIEW, přestože nebyly využity veškeré dostupné funkce z knihovny SoftMotion. V následující kapitole je dále diskuze využití průmyslového robotu.

Závěrem lze tedy konstatovat, že zadané cíle práce byly naplněny a poskytnutá řešení řízení servomotorů splňují stanovené požadavky.



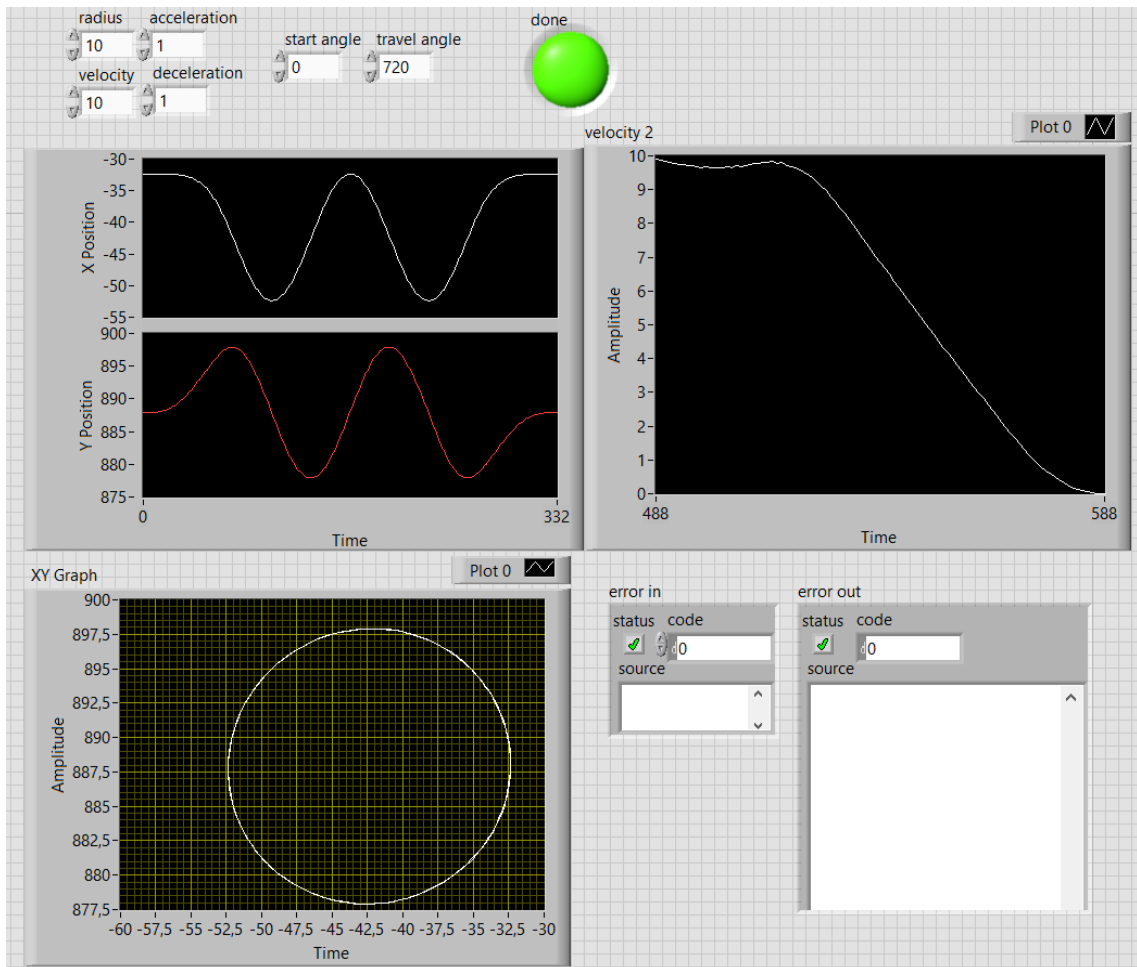
# Seznam literatury

- [1] POLIAK, František, Viliam FEDÁK a Ladislav ZBORAY. 1987. *Elektrické pohony*. 1. vyd. Bratislava: Alfa.
- [2] DIBLÍK, Martin. *Elektrické pohony*. (přednáška) Liberec: TUL, 7. 1. 2014
- [3] KULE, Lumír. 1983. *Technika elektrických pohonů*. 1. vyd. Praha: Státní nakladatelství technické literatury,
- [4] SKALICKÝ, Jiří. 2008. *Elektrické servopohony*. Brno: VUT. 84 s.
- [5] SIEMENS. 2016. *Jednoosé servo Sinamics V90* [online]. [cit. 2016-4-28].  
Dostupné z: <http://stest1.etnetera.cz/ad/current/index.php?ctxnh=2d24cc7562>
- [6] OMRON. 2016. *R88M-K, R88D-KN[/]-ECT* [online]. [cit. 2016-4-28].  
Dostupné z: <https://www.ia.omron.com/products/family/2644/feature.html>
- [7] MITSUBISHI ELECTRIC. 2016. *Série MELSERVO-J3* [online].  
[cit. 2016-4-29]. Dostupné z:  
[https://cz3a.mitsubishielectric.com/fa/cs/products/drv/servo/items/mr\\_j3/](https://cz3a.mitsubishielectric.com/fa/cs/products/drv/servo/items/mr_j3/)
- [8] KOLLMORGEN. 2013. *AC Synchronous Servo Motors – AKM<sup>TM</sup> Series* [online]. [cit. 2016-4-22]. Dostupné z:  
<http://www.kollmorgen.com/en-us/products/motors/servo/akm-series/akm-series-ac-synchronous-motors/ac-synchronous-servo-motors/>
- [9] NATIONAL INSTRUMENTS. 2016. *About NI* [online]. [cit. 2016-4-15].  
Dostupné z: <http://www.ni.com/company/about-ni/>

- [10] KRETSCHMEROVÁ, Lenka, Jaroslav VLACH. 2014. *Programování v LabVIEW v příkladech*. Liberec: TUL. Skriptum. ISBN: 978-80-7372-167-2.
- [11] MARTINI, Silvano. *Řízení a synchronizace elektrických pohonů v NI LabVIEW*. Brno: VUT, 2015. 41 s. Bakalářská práce.
- [12] NATIONAL INSTRUMENTS. 2016. *NI cRIO-9067* [online]. [cit. 2016-4-20]. Dostupné z: <http://sine.ni.com/nips/cds/view/p/lang/cs/nid/212806>
- [13] RAVEO. 2016. *AKD SERVOZESILOVÁČ* [online]. [cit. 2016-4-21]. Dostupné z: <http://www.raveo.cz/AKD-servozesilovac>
- [14] NATIONAL INSTRUMENTS. 2014. *AKD Servo Drives and Brushless Servo Motors With Smart Feedback Devices*. Manuál, 14 s. Dostupné z: <http://www.ni.com/datasheet/pdf/en/ds-280>
- [15] NATIONAL INSTRUMENTS. 2010. *Getting Started with AKD EtherCAT<sup>®</sup> Drives and the LabVIEW<sup>™</sup> NI SoftMotion<sup>™</sup> Module*. Manuál, 24 s. Dostupné z: <http://www.ni.com/pdf/manuals/375592b.pdf>
- [16] NATIONAL INSTRUMENTS. 2013. *Vision Guided Robotics* [online]. [cit. 2016-4-25]. Dostupné z: <http://www.ni.com/white-paper/10607/en/>

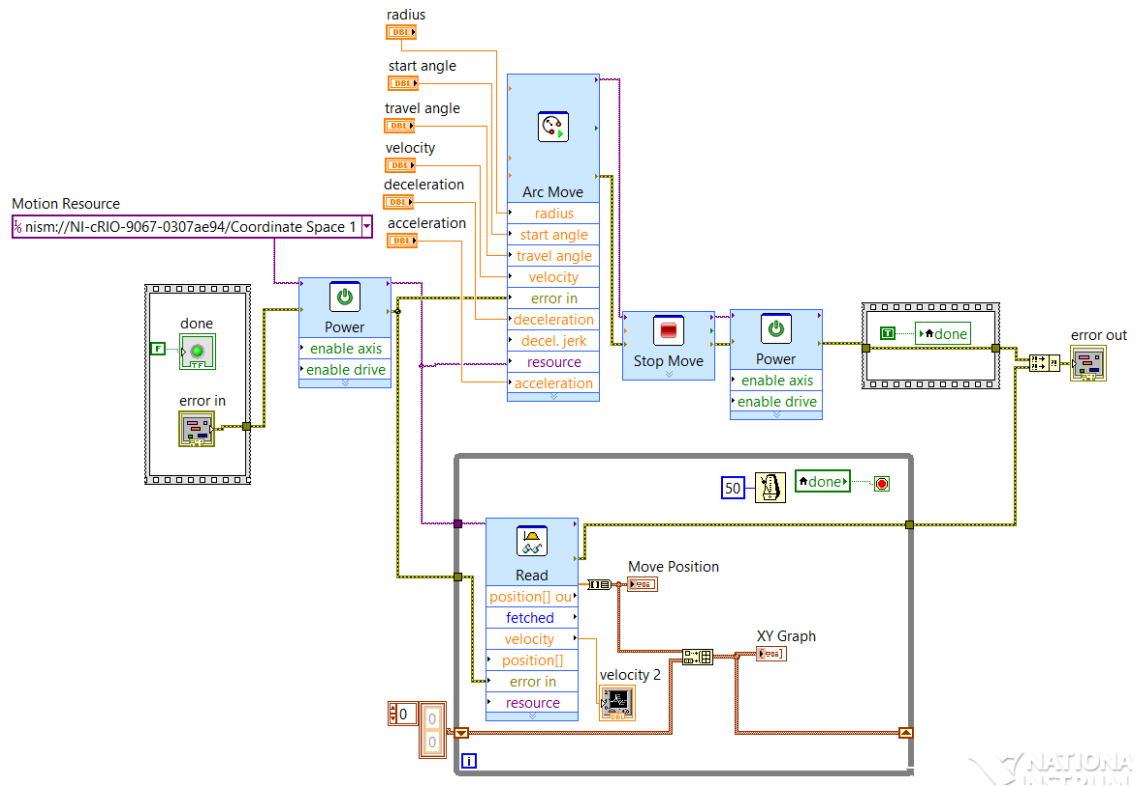
# Přílohy

## 6.1 Příloha A



Obrázek 23 - Front panel aplikace za použití funkce Arc Move

## 6.2 Příloha B



Obrázek 24 - Blokový diagram aplikace za použití funkce Arc Move