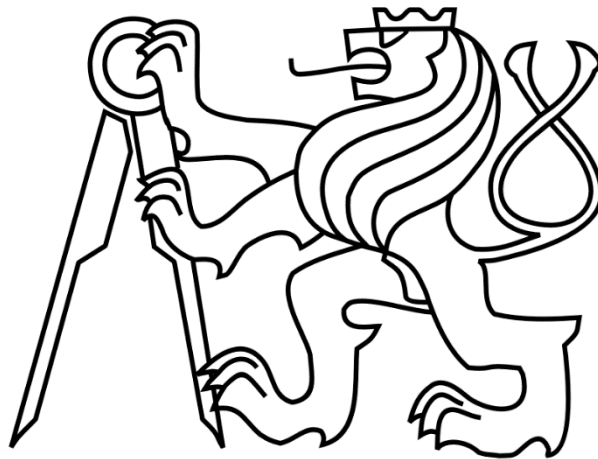


ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

Ústav výrobních strojů a zařízení



# Diplomová práce

Stand pro výuku servopohonů

*Bc. Natan Ber*

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Ber** Jméno: **Natan** Osobní číslo: **483235**  
Fakulta/ústav: **Fakulta strojní**  
Zadávající katedra/ústav: **Ústav výrobních strojů a zařízení**  
Studijní program: **Robotika a výrobní technika**  
Specializace: **Výrobní technika**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Stand pro výuku servopohonů**

Název diplomové práce anglicky:

**Stand for teaching servo drives**

Pokyny pro vypracování:

Popis tématu: Hlavním cílem je navrhnout pracoviště (stand) pro výuku servopohonů. Základ standu může vycházet ze dvou servomotorů, které budou doplněné o setrvačníky a mechanicky spojené poddajnou hřídelí. Stand by měl pomoci předat studentům potřebné základní znalosti o dynamickém chování pohybových os a o interakci pohonů s mechanickou strukturou. Osnova práce: Rešerše vhodných operačních systémů reálného času a jejich vývojových prostředí, přehled řešení firmy Beckhoff (a dalších), kompatibility s vývojovým prostředím Matlab atp.; Rešerše obdobných řešení (standů) dostupných na trhu. Návrh konstrukce standu (vč. potřebných výpočtů) a výrobní dokumentace. Tvorba řídicího projektu Beckhoff TwinCAT, implementace zpětnovazebního řízení vyvinutého ve zvoleném prostředí (např. Matlab). Tvorba uživatelského rozhraní a zajištění možnosti realizovat potřebná měření vč. sběru dat. Tvorba dokumentace. Rozsah grafické části: Výkres sestavy. Kompletní výrobní výkresy (vyřáběných dílců). Doporučený rozsah textové části: 60 - 80 stran.

Seznam doporučené literatury:

SOUČEK, Pavel. Servomechanismy ve výrobních strojích. Vydavatelství ČVUT, 2004. ISBN 80-01-02902-6  
SOUČEK, Pavel. Vložené převody v pohonech NC strojů a robotů. 2019.

Jméno a pracoviště vedoucí(ho) diplomové práce:

**Ing. Lukáš Novotný, Ph.D. ústav výrobních strojů a zařízení FS**

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

**Ing. Jan Ferkl ústav výrobních strojů a zařízení FS**

Datum zadání diplomové práce: **14.04.2023** Termín odevzdání diplomové práce: **24.07.2023**

Platnost zadání diplomové práce: **24.09.2023**

Ing. Lukáš Novotný, Ph.D.  
podpis vedoucí(ho) práce

doc. Ing. Petr Kolář, Ph.D.  
podpis vedoucí(ho) ústavu/katedry

doc. Ing. Miroslav Španiel, CSc.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

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

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta

## **Prohlášení**

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

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

V Praze dne 20. 7. 2023

.....

podpis

# Poděkování

Rád bych poděkoval vedoucímu diplomové práce Ing. Lukáši Novotnému Ph.D. za odborné rady a ochotu při konzultacích. Poděkování patří také konzultantovi Ing. Janu Ferklovi za vstřícnost a zpětnou vazbu, Ing. Jiřímu Stloukalovi z firmy Beckhoff a ostatním kolegům z Ústavu výrobních strojů a zařízení za nápady a spolupráci při řešení praktické problematiky. V neposlední řadě bych rád poděkoval mé rodině za podporu během tvorby diplomové práce.

# Anotace

Autor:	Bc. Natan Ber
Název DP:	Stand pro výuku servopohonů
Rozsah práce:	97 str., 50 obr., 2 tab.
Školní rok vyhotovení:	2023
Škola:	ČVUT v Praze – Fakulta strojní
Ústav:	Ú12135 – Ústav výrobních strojů a zařízení
Vedoucí bakalářské práce:	Ing. Lukáš Novotný, Ph.D.
Konzultant:	Ing. Jan Ferkl
Zadavatel:	ČVUT FS, Ú12135
Využití:	Návrh pracoviště pro výuku servopohonů se zaměřením na průmyslové použití
Klíčová slova:	ladění pohonů, měnič, servozesilovač, vlastní frekvence, výrobní stroje, obráběcí stroje, PLC, stand, Beckhoff, TwinCAT, Matlab
Anotace:	Diplomová práce se zabývá návrhem pracoviště pro výuku servopohonů. Součástí práce je rešerše vhodných operačních systémů reálného času a rešerše rotačních standů. Praktická část obsahuje návrh konstrukce dvouhmotového standu spolu s tvorbou řídicího projektu TwinCAT.

# Annotation

Author:	Bc. Natan Ber
Title of diploma thesis:	Stand for teaching servo drives
Extent:	97 p., 50 fig., 2 tab.
Academic year:	2023
University:	CTU in Prague – Faculty of Mechanical Engineering
Department:	Ú12135 – Department of Production Machines and Equipment
Supervisor:	Ing. Lukáš Novotný, Ph.D.
Consultant:	Ing. Jan Ferkl
Submitter of the Theme:	Faculty of Mechanical Engineering, Ú12135
Application:	Design of a stand for teaching servo drives with a focus on industrial applications
Key words:	servo drive, servo amplifier, drive tuning, natural frequency, production machines, machine tool, PLC, stand, Beckhoff, TwinCAT, Matlab
Annotation:	The master's thesis deals with the design of a stand for teaching servo drives. The thesis includes research of suitable real-time operating systems and research of rotary stands. The practical part contains the design of a two-mass stand along with the development of the control project using TwinCAT.

# Obsah

1	Úvod.....	10
2	Cíl práce .....	11
3	Možnosti řízení servopohonů .....	12
3.1	Operační systémy reálného času na stolním PC .....	13
3.1.1	Beckhoff TwinCAT 3 [1] .....	14
3.1.2	Matlab Simulink RT .....	17
3.1.3	REXYGEN .....	19
3.1.4	LabVIEW Real-Time .....	20
3.2	Vývojová prostředí a jejich propojení s RTOS.....	22
3.2.1	TwinCAT 3 Target for Simulink .....	22
3.2.2	Simulink PLC Coder .....	24
3.2.3	TwinCAT 3 Interface for MATLAB/Simulink.....	25
3.2.4	TwinCAT 3 a LabVIEW .....	26
3.2.5	Propojení nástroje REXYGEN s PLC a Simulinkem .....	27
3.2.6	LabVIEW Model Interface Toolkit.....	27
3.2.7	Beckhoff Motion Control a Drive Manager .....	28
3.2.8	NI SoftMotion .....	29
3.2.9	FPGA – programovatelné hradlové pole .....	30
3.3	Rešerše standů se zaměřením na řízení rotačních pohonů .....	31
3.3.1	Edibon SERIN/CA .....	31
3.3.2	Googoltech GSMT2014.....	33
3.3.3	Flexible two-mass motion system .....	34
3.3.4	Testovací stand z publikace Hoffmanna a kol. ....	36
3.3.5	Motion Control Demonstrator .....	38
4	Shrnutí rešerše.....	39
5	Návrh koncepčních variant pracoviště .....	40
5.1	Varianta 1 – LabVIEW s pohony Kollmorgen .....	41
5.2	Varianta 2 – REXYGEN s pohony Nanotec a Maxon .....	42
5.3	Varianta 3 – Beckhoff s vlastními pohony .....	44
5.4	Porovnání variant a výběr vhodného řešení .....	46
6	Návrh standu .....	49
6.1	Výpočet modálních vlastností pohonu .....	49
6.2	Návrh setrvačníků a spojky .....	52
6.2.1	Návrh spojky .....	53
6.2.2	Návrh setrvačnicku .....	55
6.3	Kontrolní výpočty .....	57
6.3.1	Ověření modálních vlastností .....	57

6.3.2	Kontrola svěrného spojení.....	57
6.3.3	Kontrola na otlačení .....	58
6.4	Konstrukce nosné struktury standu .....	59
6.4.1	Rám.....	60
6.4.2	Chlazení .....	62
6.5	Výkonová elektronika .....	63
6.5.1	Měniče EL7221 .....	63
6.5.2	Periferie a napájecí zdroje .....	64
6.6	Bezpečnost pracoviště.....	65
6.6.1	Ochrana obsluhy.....	65
6.6.2	Ochrana elektrického vybavení .....	65
7	Návrh řídicího projektu.....	66
7.1	Instalace Beckhoff TwinCAT .....	66
7.2	Základní nastavení TwinCAT projektu .....	67
7.3	Struktura řídicího projektu .....	69
7.4	Naprogramované funkční bloky .....	71
7.4.1	Správa os – FB_ENABLE_AXIS a FB_AXIS_STATUS_CTRL.....	72
7.4.2	Měření Bode plot – FB_BODE.....	73
7.4.3	Funkce pro pohyby os – FB_MOVEMENT.....	77
7.4.4	Funkce pro skok momentu – FB_STEP_TORQUE .....	78
7.5	Implementace řídicího algoritmu ze Simulinku.....	79
7.5.1	Simulink PLC Coder .....	80
7.5.2	TwinCAT 3 Target for Simulink .....	81
7.6	HMI projekt.....	83
7.7	Realizace výukového standu .....	86
8	Závěr .....	87
	Seznamy .....	89



## Seznam použitých veličin

Označení	Legenda	Jednotka
$\beta$	úhel boku profilu závitu	°
$\gamma$	úhel stoupání závitu	°
$\rho_{ocel}$	hustota oceli	kg/m <sup>3</sup>
$\sigma_{kt}$	dovolené tahové napětí	MPa
$\tau_d$	dovolené napětí pro krut	MPa
$\varphi$	třecí úhel	°
$\omega^*$	antirezonance	rad/s
$\Omega$	vlastní frekvence	rad/s
$d_1$	velký průměr závitu	mm
$d_2$	střední průměr závitu	mm
$d_3$	malý průměr závitu	mm
$d_{hlavy}$	průměr hlavy šroubu	mm
$d_{spojka}$	vnější průměr spojky	mm
$d_{min}$	Minimální průměr poddajné hřídele	mm
$F$	síla	N
$f$	součinitel tření	-
$f_{hr}$	frekvence filtru horní propust	Hz
$G$	modul pružnosti ve smyku	MPa
$g$	tíhové zrychlení	m/s <sup>-2</sup>
$H_1$	nosná délka závitu	mm
$h_{závit}$	hloubka závitu	mm
$J$	setrvačnost	kgm <sup>2</sup>
$k_{bezKR}$	bezpečnost v krutu	-
$k_{svěrné}$	bezpečnost svěrného spojení	-
$k_t$	torzní tuhost	Nm/rad
$l$	délka poddajné hřídele	mm
$M_k$	nominální krouticí moment motoru	Nm
$M_{kk}$	utahovací moment šroubů setrvačnicku	Nm
$M_{ks}$	moment šroubů setrvačnicku	Nm
$M_{tm}$	třecí moment šroubů setrvačnicku	Nm
$p_0$	plošné otláčení	MPa
$p_{hs}$	tlak v dosedací ploše šroubu	MPa
$p_z$	tlak v závitech	MPa
$p_{závit}$	stoupání závitu	mm
$Q$	svěrná síla	N
$r$	poloměr	mm
$S_3$	minimální průřez šroubu	mm <sup>2</sup>

# 1 Úvod

Servomechanismus je z definice automatické zařízení pro vlečnou regulaci polohy a jejích derivací. Princip činnosti spočívá ve sledování regulační odchylky, tedy rozdílu vstupní a výstupní veličiny, a snaze tuto odchylku minimalizovat. V současnosti jsou kvůli vysokému tlaku na účinnost a flexibilitu preferovány servomechanismy elektrické, obvykle rotační. Tvoří je elektromotor a určitá forma odměřování, zpravidla polohy. Servopohony nalézají široké uplatnění v mnoha aplikacích. Často se jedná o klíčové části výrobních strojů, proto je zde kladen vysoký důraz na stabilitu řízení. V průběhu času byla testována řada řešení, nejvíce se osvědčila kaskádní regulace. Sestává z proudové, rychlostní a polohové zpětné vazby. První dvě smyčky obsahují proporcionálně-integrační regulátor, v polohové smyčce se nachází pouze proporcionální regulátor. Naladění regulačních konstant má klíčový vliv na dynamickou poddajnost celé soustavy.

Stand pro výuku servopohonů má za úkol demonstrovat interakci servopohonů s mechanickou strukturou a dynamické chování pohybových os. O řízení servopohonů, sledování stavu při provozu a zadávání požadované polohy se stará řídicí systém. V úvodu práce bude prezentována rešerše operačních systémů reálného času vhodných pro řízení servopohonů. Důraz bude kladen na cenovou dostupnost softwaru i hardwaru. Protože se u standu předpokládá laboratorní použití, bude v práci popsána vzájemná kompatibilita vývojových prostředí. Ve druhé části rešerše budou uvedeny komerčně nabízené výukové standy a standy popsané v odborných publikacích.

Následuje výběr vhodného řídicího systému a koncepce pracoviště. V praktické části bude proveden konstrukční návrh standu včetně tvorby výkresové dokumentace a výběru elektronických komponent. Pro vybranou platformu bude vytvořen a v práci popsán řídicí projekt, jehož součástí bude ovládání pohonů, měření a sběr dat. Zahrnut bude také implementovaný vlastní řídicí algoritmus.

## 2 Cíl práce

Cílem práce je navrhnout stand pro výuku servopohonů. Pracoviště má být tvořeno dvojicí servomotorů spojených poddajnou hřídelí. Pro zvýšení setrvačnosti motorů mohou být použity přídavné setrvačníky. Stand by měl sloužit k demonstraci chování pohybových os strojů a měl by názorně předvést interakci pohonů s mechanickou strukturou.

V rešeršní části budou představeny vhodné operační systémy reálného času, příslušná vývojová prostředí a jejich vzájemná kompatibilita, primárně s programem Matlab. Bude zde také provedena rešerše obdobných řešení dostupných na trhu.

V navazující části práce bude navržena konstrukce standu včetně potřebných výpočtů a výkresové dokumentace. Ovládání standu bude zajištěno vytvořeným řídicím projektem v Beckhoff TwinCAT, jehož součástí bude implementované zpětnovazební řízení vyvinuté v programu Matlab Simulink. Součástí bude rovněž uživatelské rozhraní pro možnosti spuštění a monitorování potřebných měření včetně sběru dat.

### 3 Možnosti řízení servopohonů

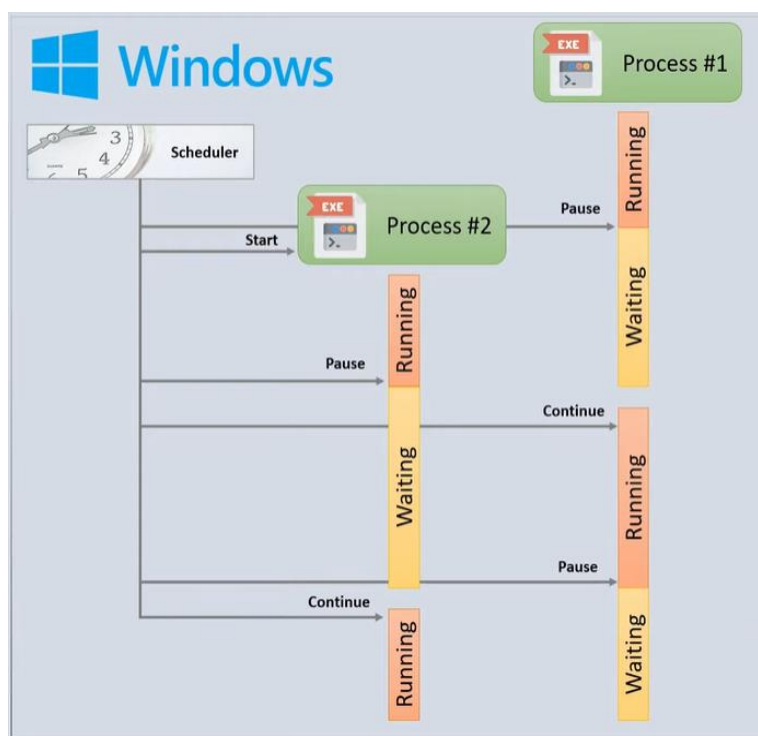
Servopohon je obvykle tvořen motorem, měničem (označovaným také jako servozesilovač) a řídicím systémem. Řídicí systém se stará o tvorbu časového signálu požadované polohy či rychlosti. Tento signál může tvořit například na základě interpretace NC programu, který interpoluje do časové oblasti. Součástí řídicího systému může být rovněž kinematická transformace vstupních dat nebo omezení signálu limitními hodnotami. Vytvořený časový průběh je poté posílán do měniče, který je propojený s motorem. Měnič převádí požadovaný proud do motoru na sinusový signál napětí. Dosahuje toho pulzní šířkovou modulací (pulse width modulation – PWM), což v praxi znamená spínání a rozpínání výkonových tranzistorů. Řízení servomotoru zajišťuje kaskádní regulační obvod. Proudová regulační smyčka je tvořena proporcionálně-regulačním (PI) regulátorem a je téměř výhradně součástí frekvenčního měniče. Rychlostní smyčka je rovněž tvořena PI regulátorem a může být uzavírána jak ve frekvenčním měniči, tak v řídicím systému. To samé platí pro polohovou regulační smyčku s jediným rozdílem – je tvořena pouze proporcionálním (P) regulátorem.

Řídicí systém je tedy klíčovou součástí servopohonu. Jedná se o výpočetní zařízení, na kterém běží operační systém a řídicí program. Pro bezpečné a jasně definované řízení se musí jednat o operační systém reálného času, neboli real-time OS. V úvodu rešerše jsou popsány systémy různých výrobců vhodné pro řízení pohonů. Představena jsou také vývojová prostředí a možnosti jejich vzájemných propojení. Při výběru byl kladen důraz na cenu s ohledem na plánované použití standu ve výuce. Také zde bylo bráno v potaz laboratorní prostředí, ve kterém bude stand provozován. Ve druhé části rešerše jsou představeny komerčně dostupné standy pro výuku servopohonů. Jsou zde rovněž uvedena pracoviště z odborné literatury, používaná pro laboratorní testování a verifikaci řídicích algoritmů.

### 3.1 Operační systémy reálného času na stolním PC

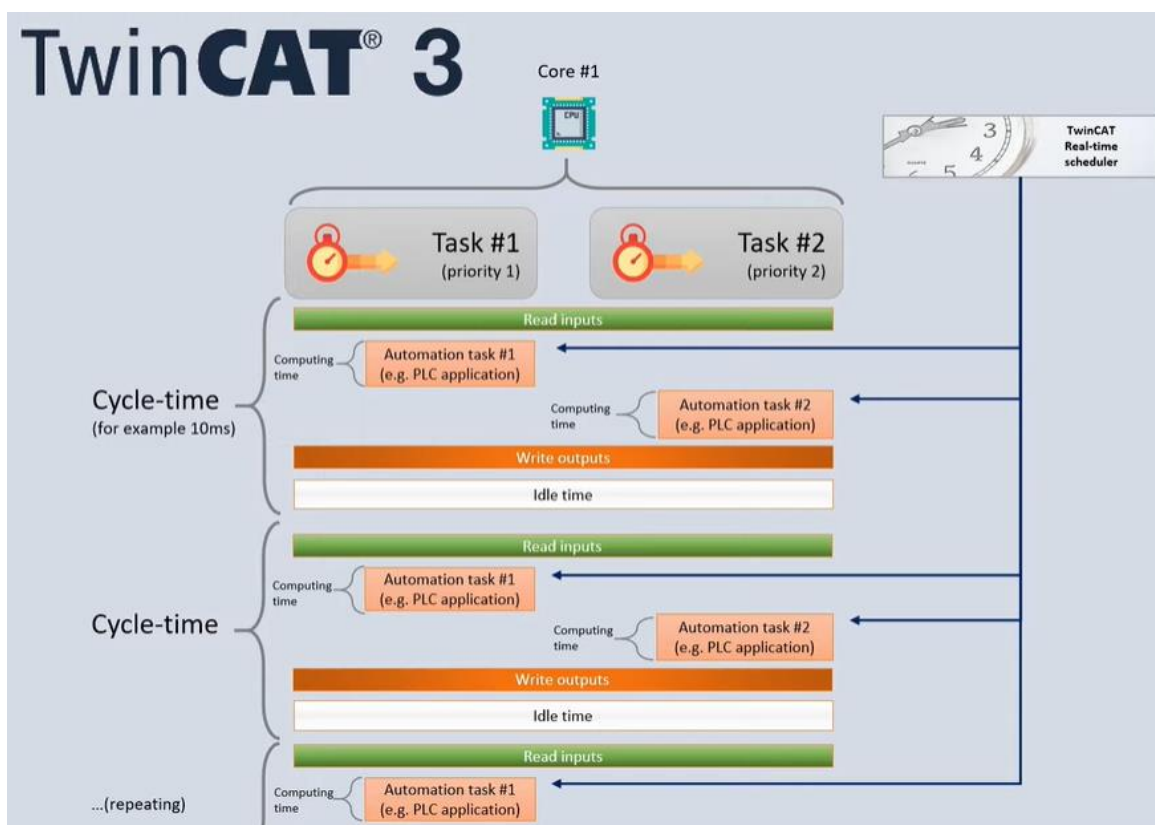
Dle normy DIN 44300 jsou operační systémy reálného času (Real Time Operating Systems – RTOS) takové systémy, u kterých správnost výstupu nezávisí pouze na výsledku logických výpočtů, ale také na čase, ve kterém byly výsledky zpracovány. [1] RTOS systém spouští zdrojový kód v cyklech, čas tohoto cyklu může být pevně daný, např. 10 ms, nebo může být proměnný. U druhé varianty každý cyklus započne ihned po dokončení výpočtu v předešlém cyklu, je zde ovšem jasně definovaný maximální čas cyklu. Při překročení tohoto limitu může dojít k chybě, v kritickém případě až ke kolapsu systému. [2] Pro tyto případy by měla být v řídicím systému nastavena funkce pro bezpečné odstavení systému.

Chování systému musí být deterministické, pro stejný vstup musí být vždy stejný výstup, každý aktuální a následující krok výpočtu je jasně definován. Níže (Obrázek 3-1) je možné vidět princip fungování běžného operačního systému. Plánovač rozhoduje, na kterém vlákne procesoru poběží jaký proces. Není neobvyklé střídavé spouštění několika procesů najednou a přepínání mezi nimi, což vyústí v rozdílné časy dokončení výpočetních úloh.



Obrázek 3-1 - Nedeterministické chování běžného operačního systému [3]

Na rozdíl od běžného operačního systému je u RTOS pevně stanoveno pořadí úloh a úlohy se vykonávají jedna po druhé (Obrázek 3-2). Obvykle proběhne nejprve čtení vstupů, následně samotný program a na závěr se zapíše výstupy. Na obrázku je příklad řešení TwinCAT 3 od Beckhoffu s pevně nastaveným časem cyklu 10 ms, z tohoto důvodu procesor po výpočtu čeká na začátek dalšího cyklu (Idle time). [3]



Obrázek 3-2 – Operační systém reálného času [3]

Operační systémy reálného času mohou běžet i v rámci běžného operačního systému, v dalších podkapitolách budou popsány dostupné varianty RTOS, které jsou zároveň vhodné pro ovládání pohonů (motion control) a dále princip fungování těchto systémů uvnitř běžného OS.

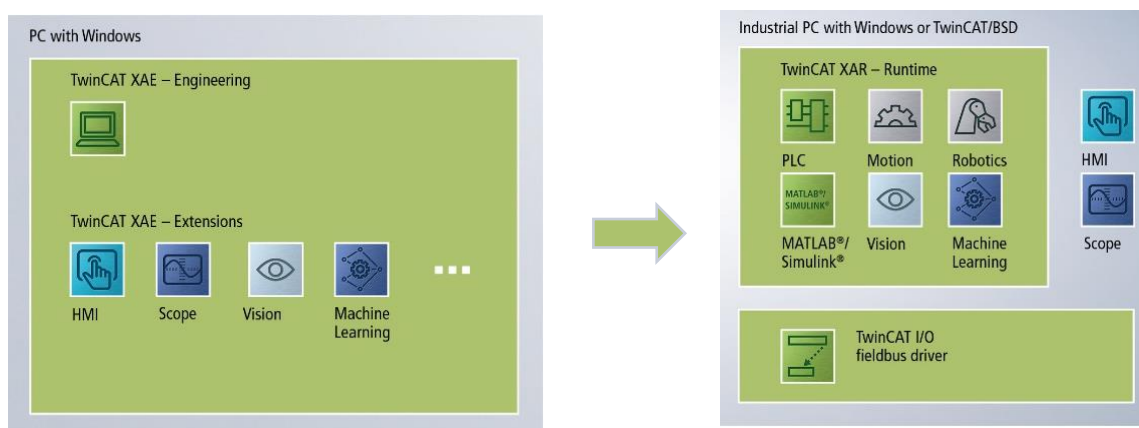
### 3.1.1 Beckhoff TwinCAT 3 [1]

Německá společnost Beckhoff je zavedeným výrobcem otevřených automatizačních systémů založených na platformě PC. Kromě vývoje průmyslových počítačů a programovatelných logických kontrolérů (PLC) poskytuje společnost také komponenty I/O, komponenty průmyslových sběrnic, technologie pohonů

a automatizační software, který vychází z Windows. Nazývá se TwinCAT (The Windows Control and Automation Technology) a první verze byla uveřejněna roku 1996, nyní je na trhu již třetí generace. Pomocí softwaru TwinCAT je možné transformovat běžný stolní počítač, který vyhovuje výkonnostním požadavkům, na řídicí systém reálného času. Tento systém je následně schopen spouštět PLC, NC či CNC programy. TwinCAT je možné modulárně rozšiřovat, případně díky otevřenosti systému samotná rozšíření vytvářet.

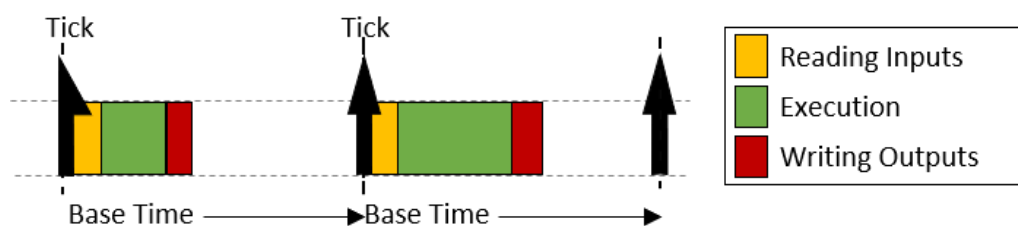
Samotný TwinCAT tvoří dvě části – XAE a XAR. První zmiňovaná zkratka značí eXtended Automation Engineering. Jedná se o vývojové prostředí, ve kterém je hardware programován a konfigurován. Toto vývojové prostředí je zcela zdarma. Součástí prostředí jsou nástroje pro debugging a diagnostiku řídicího systému. Programování může probíhat dle normy IEC 61131-3, například s využitím jazyka Ladder Diagram, Function Block Diagram nebo Structured Text. Další možností je psát program v jazyku C/C++ nebo v prostředí Matlab Simulink. Propojení TwinCATu s dalšími vývojovými prostředím bude rozepsáno v kapitole 3.2. [4]

XAR, neboli eXtended Automation Runtime, je běhové prostředí, anglicky označované run-time environment. Zde může být v reálném čase odbavován program, který obsluhuje připojené vstupy a výstupy. Toto prostředí je provozováno nezávisle na operačním systému výpočetního zařízení. OS ovšem vždy běží v pozadí a je na něm možné spouštět další programy, vizualizaci, logování dat a tak dále. Výpočtem a vizualizací naměřených dat v operačním systému není zatěžováno real-time prostředí, běžící na oddělené části jádra (kernelu). Spojitost mezi XAE a XAR je naznačena níže (Obrázek 3-3).



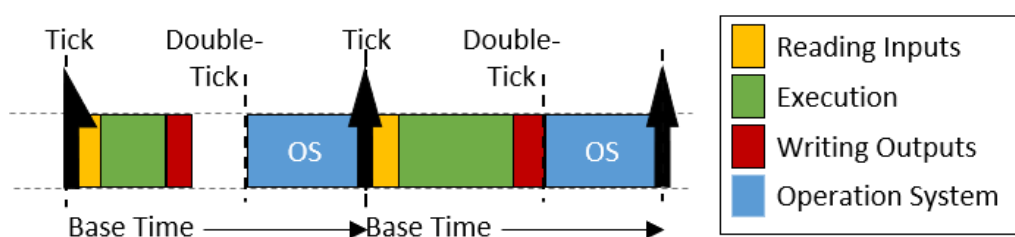
Obrázek 3-3 - TwinCAT 3 [4]

XAR je zprolatněný dle výkonosti hardwaru, na kterém je provozován. Umožňuje ovšem opsáním kódu vygenerovat sedmidenní trial licenci, v rámci které je většina funkcí plně dostupných. Po uběhnutí tohoto období je nutné opět opsat CAPTCHA kód. Operační systém reálného času může běžet na samostatném dedikovaném procesoru nebo na procesoru spolu s běžným operačním systémem. První případ demonstruje schéma uvedené v dokumentaci společnosti Beckhoff (viz Obrázek 3-4). [1] Base Time stanovuje čas cyklu, žlutě je označen čas pro čtení vstupů, zeleně průběh programu a červeně zápis do výstupů. Zbylý čas procesor čeká.



Obrázek 3-4 - RTOS na izolovaném jádru procesoru [1]

Ve druhém případě běží na jádře jak RTOS, tak běžný operační systém (Obrázek 3-5). V rámci cyklu (base time) dojde nejprve ke čtení vstupů, následně proběhne program a zapíše se výstupy. Poté se TwinCAT přepne z real-time operačního systému do běžného operačního systému, zde Windows 10. Přepnutí z RTOS do OS je na obrázku znázorněno přerušovanou čarou s popiskem Double-Tick (Obrázek 3-5). V tomto režimu tedy vždy proběhne PLC program v předem definovaném čase cyklu, ve zbylém čase využívá výpočetní výkon OS. Toto řešení přináší ovšem úskalí v podobě menšího času na provedení PLC programu a větší náročnosti na výkon procesoru, protože je nutné přepínat mezi systémy a celý běh musí být dokončen v definovaném čase. Vhodnější je nejprve přečíst vstupy, následně zapsat výstupy z předešlého cyklu a až poté spustit samotný program. Docílí se tím stejného času zápisu výstupů v každém cyklu a tím i předvídatelnějšího chování celého systému.



Obrázek 3-5 - RTOS na jádře spolu s OS [1]



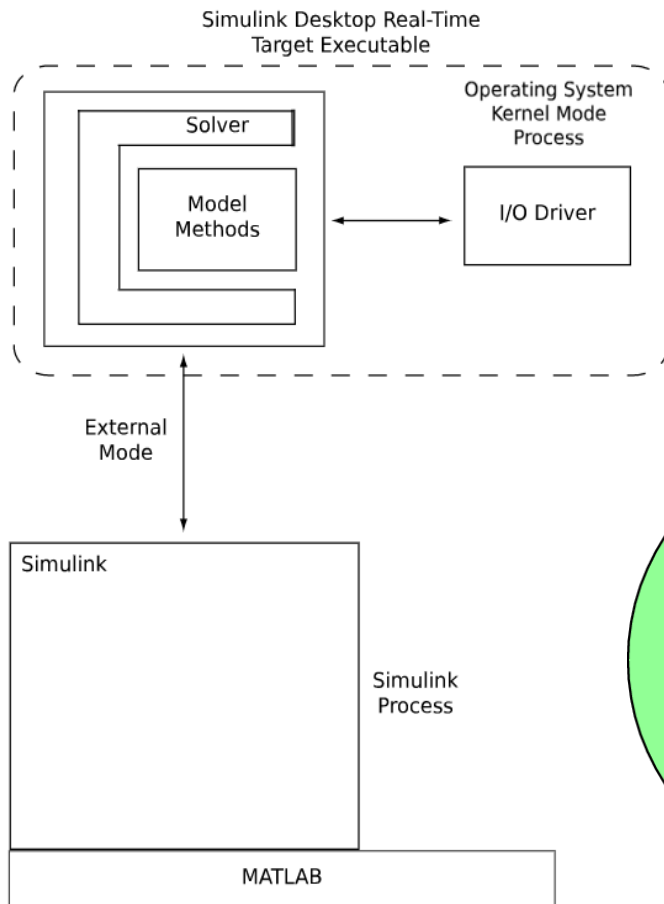
Pro korektní deterministický real-time běh je nutné, aby měl TwinCAT přímý přístup do jádra operačního systému (ring 0, Obrázek 3-6). [5] Procesy na jádře mají přímý přístup k hardwaru, běží zde Windows kernel, obsluha vstupů a výstupů počítače, síťové porty a obsluha periférií. Běžné aplikace se spouští v okruhu Ring 3, který má mnohem více omezený přístup k hardwaru, což funguje také jako ochrana pro případ chyb v aplikacích, které by mohly shodit celý systém. Běží zde všechny procesy, virtuální operační paměť a uživatelské prostředí. Tato oblast je též vhodná pro tvorbu vizualizace, zpracování dat a ovládací panely HMI.

### 3.1.2 Matlab Simulink RT

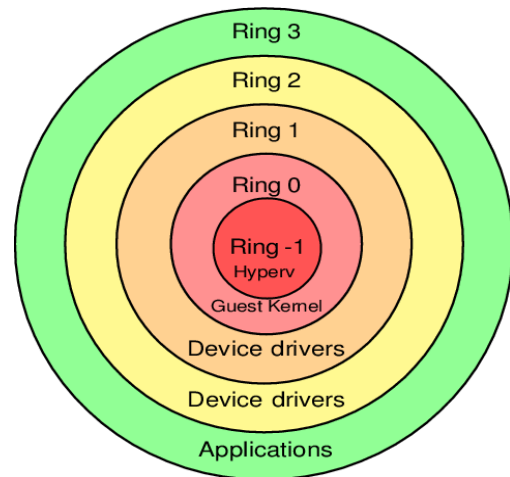
Podobně jako TwinCAT může i Matlab běžet v reálném čase v kombinaci s běžným stolním počítačem, a to ve dvou variantách. První varianta spouští RT program na externím zařízení výrobce Speedgoat a nazývá se Simulink Real-Time (SLRT). Do verze R2020b bylo možné používat cílová zařízení (target computer) od různých společností, například dSpace nebo National Instruments. To již bohužel není v novějších verzích možné. Speedgoat nabízí celou řadu zařízení, od Performance řady určené k náročným výpočtům v laboratoři, přes Mobile řadu pro práci v terénu až po modulární Tailored Rack-System s kartami na míru. [6] Hlavní výhodou externího zařízení od Speedgoat je vysoký výkon a vzorkovací frekvence, možnost simulování kontrolérů pro HiL simulace a rapid prototyping, nevýhodou je vysoká cena. [7]

Druhá varianta Simulink Desktop Real-Time (SLDRT) běží na běžném počítači spolu s vývojovým prostředím. Matlab poskytuje real-time kernel, který působí v prvním okruhu privilegií operačního systému (Obrázek 3-6), stejně jako TwinCAT kernel. Jeho součástí jsou I/O bloky pro Simulink. Simulink Desktop Real-Time běží v reálném čase paralelně se Simulinkem běžícím v normálním módu, komunikace je zprostředkována právě přes tyto bloky. [8] Díky tomu je možné sledovat aktuální hodnoty signálů v modelu, logovat signály a vytvářet interaktivní ovládací prvky. Nevýhodou tohoto řešení je výrazně nižší vzorkovací frekvence při porovnání se SLRT, Matlab udává frekvence do 1 kHz. Řešením může být generování C/C++ kódu pomocí Simulink Coder, Matlab poté běží v externím módu (external mode), který je možné provozovat se vzorkovací frekvencí až 20 kHz.

Obrázek 3-7 zobrazuje Matlab a Simulink běžící v normálním módu ve Windows a následnou komunikaci se zkompilevaným modelem v real-time režimu na jádře procesoru, ve kterém běží také obsluha vstupů a výstupů modelu.



Obrázek 3-7 – Simulink Desktop Real-Time v externím módu [8]



Obrázek 3-6 – Okruhy privilegií pro procesory s architekturou x86 [5]

Pro připojení vstupů k počítači je možné použít např. PCI-E karty od firmy Humusoft nebo dSPACE. SLDRT dokáže běžet pouze na jednom jádře výhradně na počítači, kde je spuštěn Simulink. Nepodporuje integraci programovatelného hradlového pole FPGA nebo použití v terénu na externím real-time zařízení v režimu „stand alone mode,“ například při testování řídicích jednotek v automobilech.

### 3.1.3 REXYGEN

REXYGEN je otevřený nástroj pro konfiguraci a realizaci řídicích algoritmů, měření, modelování systémů a pro mnohé další aplikace, vyvíjený plzeňskou společností REX Controls s. r. o. Jeho součástí je vývojové prostředí REXYGEN Studio, podporující vlastní grafický programovací jazyk velice podobný Simulinku od MathWorks. Další možností programování je využití jazyka FBD (Function Block Diagram – jazyk funkčních bloků) dle standardu IEC 61131-3. K dispozici je knihovna bloků s řadou základních funkcí, jako jsou časovače, funkce pro zpracování signálu nebo PID regulátor. Další vývojářský nástroj REXYGEN HMI Designer slouží k tvorbě grafického uživatelského prostředí, pomocí kterého je možné sledovat a ovládat systém přímo na pracovní stanici.

Vytvořený program je následně nutné zkompileovat pomocí REXYGEN Compiler přímo v uživatelském prostředí, poté je kód nahrán na cílové zařízení a spuštěn. REXYGEN využívá runtime RexCore, který v případě běhu na průmyslovém počítači s operačním systémem GNU/Linux běží jako systémová úloha, v případě použití embedded controlleru může fungovat jako celý operační systém. Součástí RexCore je také webserver pro snadné propojení HMI a REST API pro sdílení dat. Spouštění programů a běh jádra lze diagnostikovat přímo pomocí REXYGEN Studio na vývojovém počítači, který je připojený k cílovému zařízení síťovým kabelem.

Software REXYGEN podporuje rozličný hardware. Pro sběr dat lze využít Raspberry Pi s rozšířením Monarco HAT. Alternativou k Monarco HAT je Pigeon PLC nebo české UniPi Neuron, opět pro platformu Raspberry Pi. Při použití počítače Raspberry Pi ovšem nelze považovat toto řešení za plně průmyslové, a to především kvůli spolehlivosti Raspberry Pi a jeho dostupnosti na trhu. Umístění do kritických částí řídicích systémů je nevhodné. Místo Raspberry Pi lze využít průmyslový počítač se systémem GNU/Linux a nástrojem REXYGEN, případně procesorový PLC modul WAGO PFC100/200 s RexCore runtime, který tvoří kompromis mezi levnými moduly založenými na Raspberry Pi a dražšími průmyslovými počítači. [9]

### 3.1.4 LabVIEW Real-Time

Společnost National Instruments (NI) přináší na trh také své real-time řešení. V rámci programu LabVIEW nabízí modul Real-Time, s jehož pomocí je možné sestavit program pro dedikovaný RT hardware, který společnost sama dodává. Dříve podporovala také Phar Lap ETS real-time operační systém, jenž umožňoval přeměnu běžného počítače na RT systém, od roku 2022 však Phar Lap kompletně ztrácí podporu softwaru společnosti National Instruments. Zároveň na rozdíl od konkurenčních řešení není možnost spustit RTOS paralelně s běžným OS, takže jsou zákazníci plně odkázáni na originální hardware. [10]

Společnost NI využívá na svých zařízeních systém postavený na open source platformě Linux. Ten nejdříve nebyl považován jako vhodný pro implementaci RTOS, ale s jeho rostoucí popularitou v embedded systémech vývojáři začali pracovat na úpravě jádra a zejména jeho deterministického chování. S příchodem patche PREEMPT\_RT se konečně podařilo zprovoznit funkční real-time systém, na jehož základě postavila společnost National Instruments operační systém NI Linux Real-Time. Z toho pramení také jeho největší výhody – kompatibilita Linuxu s ostatními zařízeními, ať už jde o možnosti síťového propojení, komunikační protokoly jako IPv6 a SNMP, nebo možnosti připojení uživatelských panelů HMI a dalších periférií. Dalším plusem je široká uživatelská základna systému Linux, která nabízí mnoho funkcí, nástrojů a návodů. Integrace C/C++ kódu je také možná, mezi další plusy patří kvalitní zabezpečení, ať už se jedná o správu uživatelských účtů a oprávnění, nebo implementaci VPN a Firewallu. [11]

NI Linux Real-Time podporuje pouze hardware společnosti National Instruments. Ta nabízí řadu platforem, vhodných pro laboratorní i průmyslové použití, všechny jsou postavené na běžné architektuře, proto je možné sdílet programy mezi jednotlivými zařízeními. PXI platformu tvoří robustní šasi, embedded controller a I/O moduly, USB konektory a gigabitový ethernet. Hlavní výhodou je použití open source modulárních karet, které nabízejí vysokou konektivitu, přenosové rychlosti a variabilitu. NI nabízí přes 600 karet, dalších téměř 1 500 karet je k dispozici od více než 70 externích dodavatelů. Zařízení jako jediné z nabídky může běžet jak na LabVIEW RT, tak na Windows 10 IoT nebo Windows 11 IoT. [12]

Mezi další populární hardware firmy National Instruments se systémem LabVIEW RT patří CompactRIO. Jedná se o odolné průmyslové výpočetní zařízení, obsahující USB, sériové a ethernet porty, integrovaný FPGA obvod a vlastní „C Series“ karty se vstupy a výstupy. Pro možnosti laboratorního testování, případně integrace controlleru do vlastní konstrukce, nabízí NI samotnou základovou desku „CompactRIO Single-Board Controller.“ Na vrcholu nabídky stojí „NI Industrial Controller“. V nejlepší konfiguraci disponuje dvoujádrovým procesorem Intel i7, 8 GB DDR3 RAM a 64 GB úložištěm. Odolná konstrukce s krytím IP67 neobsahuje větráky ani pohybující se části. Součástí je FPGA obvod a podpora průmyslových komunikačních standardů jako jsou EtherCAT, PTP a TSN.

Pro psaní RT programů je možné využít grafické vývojové prostředí LabVIEW spolu s příplatkovým modulem „Real-Time Module“ zajišťujícím kompilaci kódu pro systémy reálného času. Zároveň lze přímo z vývojového prostředí nahrávat software na RT hardware pomocí USB nebo Ethernetu a kód také za běhu debugovat pomocí funkcí highlight execution, single stepping a breakpoints. Modul podporuje většinu knihoven z LabVIEW – matematické funkce, zpracování signálu, PID regulátory a další. Také je možné psát textové skripty s využitím modulu „LabVIEW MathScript RT Module.“ Grafické vývojové prostředí je velice podobné konkurenčnímu Simulinku od MathWorks.

LabVIEW Real-Time umožňuje nastavení běhu kódu na více jádrech/vláčknech a jejich podrobnou konfiguraci. Jedno jádro může tedy běžet například izolovaně v taktu 1 kHz a provádět kritickou úlohu, druhé jádro pak může obsluhovat zbytek kódu v nižším taktu. Je zde také možné nastavovat prioritu a pořadí provedených úloh, maximální čas pro provedení smyčky a další nastavení. Defaultně provádí nastavení automaticky LabVIEW, manuální konfigurací lze ovšem zvýšit výkon programu. Pro pokročilé vícejádrové ladění je možné využít „Real-Time Trace Viewer“ a logovat s ním hodnoty bez nutnosti přerušování chodu systému, sledovat vytíženost jader a jejich přepínání nebo zaznamenávat alokaci paměti. Další alternativou ke grafickému vývojovému prostředí je psaní kódu v C/C++ pomocí open source vývojářských prostředí, například Eclipse.

## 3.2 Vývojová prostředí a jejich propojení s RTOS

V předešlé části rešerše byly popsány operační systémy reálného času pro stolní počítače, vhodné pro ovládání testovacího a výukového standu. Každý RTOS využívá vlastní vývojové prostředí (IDE), pro určité úlohy jsou ale konkurenční vývojová prostředí výrazně lepší. Tato podkapitola se bude zabývat některými z možností propojení RTOS a IDE různých výrobců, dále zde budou popsány pokročilé funkce řízení servopohonů v nativních vývojových prostředích a v posledním oddílu bude krátce uvedeno využití a programování hradlových polí FPGA. Matlab Simulink je silným nástrojem pro vývoj multidomenových simulačních modelů. Pomocí blokových diagramů lze navrhnout řídicí logiku nebo popsat fyzikální chování systému.

K urychlení a zjednodušení tvorby modelů slouží simulační nástroj Simscape, v rámci něhož jsou dostupné elektrické, mechanické, magnetické, hydraulické a plynné, termální a další knihovny. Jak již bylo zmíněno, Simulink v novějších verzích nepodporuje přímou implementaci externích real-time zařízení, výjimkou jsou zařízení dceřiné společnosti Speedgoat. MathWorks ovšem nabízí několik alternativních řešení exportování kódu ze Simulinku, nyní zde budou postupně představena.

### 3.2.1 TwinCAT 3 Target for Simulink

První varianta je určena výhradně pro propojení se zařízeními Beckhoff a nazývá se TC3 Target for Simulink. Umožňuje integraci modelů ze Simulinku přímo do vývojového prostředí TwinCATu (XAE) a to tak, že se z vytvořeného modelu vygeneruje C/C++ kód pomocí Simulink Coder a ten se následně automaticky převede na TwinCAT objekt pomocí TwinCAT 3 Target for Simulink. Tyto objekty mohou být buď TcCOM objekty, se kterými následně komunikuje zbytek programu v XAR nebo funkční bloky, které jsou součástí programu. TcCOM je zkratkou pro TwinCAT Component Object Model a je založen na standartu COM, vyvinutém Microsoftem, pro umožnění komunikace mezi různými programovacími jazyky. Beckhoff přejal část standartu COM, některé funkce následně přidal, např. sledování stavu objektu. Simulinkový model se v případě exportu jako TcCOM objekt zobrazuje v XAE jako TC BlockDiagram, který umožňuje online monitorování aktuálních hodnot, nastavování parametrů bloků a případné debugování a základní úpravu bez nutnosti opětovné kompilace z Matlabu.

Data z TwinCATu je možné po spuštění PLC zpětně odesílat do Simulinku přes ADS komunikaci, ve kterém jde také nastavovat parametry modelu stejně jako v XAE, debugovat kód a zobrazovat aktuální hodnoty veličin. [13]

TcCOM objekt má vlastní vstupy a výstupy, se kterými se zachází podobně jako s externími. Díky tomu je možné snadno provádět řadu simulací. Varianty propojení hardwaru a softwaru při vývoji budou popsány na příkladu standu pro servopohony. Model standu označuje matematický model mechanické struktury standu v prostředí MATLAB Simulink. Řídicí algoritmus je v tomto případě kaskádní regulace servomotorů. Níže jsou uvedené jednotlivé vývojové stupně:

- Model in the Loop (MiL)
  - Vytvoření modelu standu a návrh řídicího algoritmu v Simulinku
  - Umožňuje rychlé a snadné změny konstrukce (přidání setrvačnosti, převodu, změny tuhostí) a řídicího algoritmu
  
- Software in the Loop (SiL)
  - Model standu exportován do TwinCATu jako TcCOM objekt
  - Řízení probíhá pomocí TwinCAT programu (funkčních bloků) vygenerovaného ze Simulinku nebo napsaného ručně
  - Řídicí algoritmus a TcCOM model standu běží na stejném PLC
  - Verifikace implementace řídicí logiky do programovacího jazyka PLC Beckhoff
  
- Hardware in the Loop (HiL)
  - Model standu běží na samostatném PLC Beckhoff jako TcCOM objekt
  - Řídicí algoritmus implementován ve druhém PLC Beckhoff (jedná se o PLC, které bude v budoucnu obsluhovat hotový stand)
  - Propojení hardwarových vstupů a výstupů obou PLC
  - Verifikace chování koncového PLC včetně skutečných vstupů a výstupů
  - Možnost simulace chybových stavů (prasknutí hřídele, chybná data z enkodérů, ...)

- Rapid Controller Prototyping (RCP)
  - Řídicí algoritmus jako TcCOM objekt v PLC Beckhoff (nejedná se o finální implementaci ve strukturovaném textu, ale pouze o exportovaný model ze Simulinku)
  - Připojení k reálnému standu se servopohony
  - Možnost snadné úpravy řídicího algoritmu v Simulinku a změny řídicího hardwaru v případě potřeby

Pro využití TC3 Target for Simulink jsou nutné mimo standardní licence pro Matlab Simulink a Beckhoff TwinCAT 3 také dva moduly od Beckhoffu – TC3 C++/MatSim a TC3 Target For Matlab Simulink, následně další dva moduly od MathWorks – MATLAB Coder a Simulink Coder. Jedná se tedy o poměrně nákladné řešení, které může být opodstatnitelné například při zmíněných aplikacích při vývoji nového produktu a jeho testování před nasazením na trh.

### 3.2.2 Simulink PLC Coder

Další variantou exportu modelu z Matlabu je využití modulu Simulink PLC Coder, který umožňuje generování strukturovaného textu a ladder diagramu dle normy IEC 61131-3. Strukturovaný text podporuje syntax běžných vývojových prostředí, jako jsou Siemens TIA Portal, Beckhoff TwinCAT, Rockwell Automation Studio 5000 nebo PLCopen XML. Pro ladder diagram je podporován syntax užívaný ve vývojovém prostředí Rockwell Automation Studio 5000.

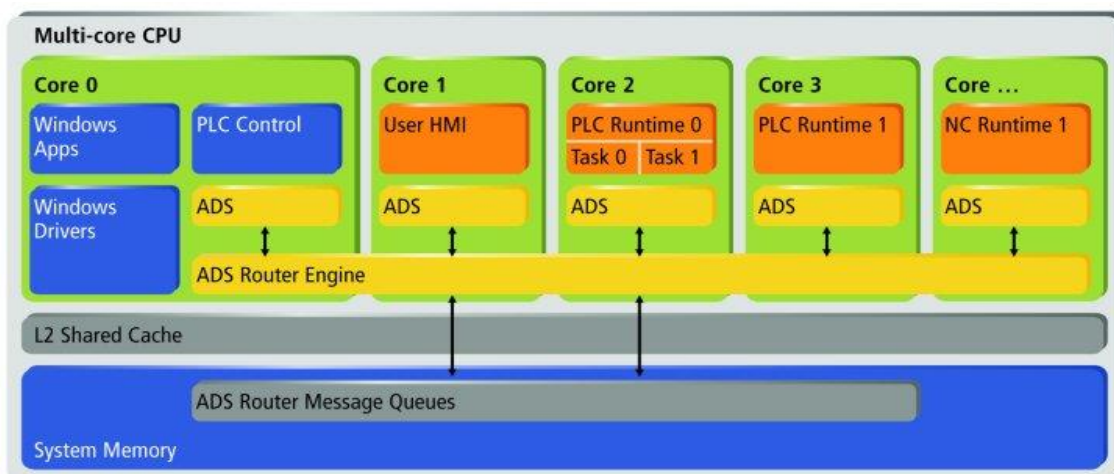
Simulink PLC Coder nabízí také nástroje pro verifikaci, optimalizaci a testování vygenerovaného kódu. V rámci optimalizace je možné snížit využívanou paměť a zvýšit celkovou efektivitu kódu. Pro verifikaci kódu je možné vytvořit testovací sadu dat pomocí funkce Test Bench Verification, která nejprve simuluje model v Simulinku, zaznamená vstupní a výstupní data subsystému a následně je exportuje do formátu pro cílové zařízení. Do cílového IDE je importován vygenerovaný kód, spuštěná sekvence vstupních dat jako při simulaci subsystému a výstupní data mohou být následně porovnána s importovanými verifikačními výstupy simulačního modelu. Porovnání dat může být provedeno automaticky, výstupem je poté zpráva, zda kód vyhověl, či nikoli.

[14]



### 3.2.3 TwinCAT 3 Interface for MATLAB/Simulink

Poslední varianta propojení Matlabu se zařízeními společnosti Beckhoff se nazývá TwinCAT 3 Interface for MATLAB/Simulink a umožňuje komunikaci mezi Matlabem, Simulinkem a TwinCAT runtime. Toto řešení je vhodné především pro software in the loop simulace, kdy se v Simulinku vytvořený model mechanické součásti propojí s řídicím algoritmem běžícím v Beckhoff runtime – XAR. Komunikace probíhá přes ADS protokol (The Automation Device Specification) vyvinutý společností Beckhoff. Protokol je využíván nejen pro výměnu dat programu, ale také pro externí přístup k hardwarovým vstupům a výstupům PLC, detekci změn statusu stroje a přístup k HMI a webovým aplikacím přes ADS web. Díky modulární architektuře TwinCATu je možné přiřadit ke každému softwarovému modulu (PLC Runtime, NC Runtime, HMI) přístupové rozhraní ADS device, jehož součástí je komunikační port. Porty komunikují s ADS routerem umožňujícím čtení a zápis dat z externích zdrojů (Obrázek 3-8).



Obrázek 3-8 Struktura ADS komunikace [1]

TwinCAT 3 Interface for MATLAB/Simulink poskytuje klienta s názvem ADS client, pomocí kterého může Matlab a Simulink komunikovat s ADS routerem a následně číst a zapisovat hodnoty do TwinCAT runtime. Dále nabízí ADS server, díky kterému může poskytovat Matlab a Simulink služby pro další ADS klienty. Propojení mezi Simulinkem a TwinCATem může být synchronní nebo asynchronní a cyklické nebo založené na událostech (event-based). Při synchronním propojení ADS klient čeká na odpověď ADS serveru předtím, než spustí požadovaný příkaz. V asynchronním režimu komunikuje ADS klient s ADS serverem bez čekání na potvrzení požadavku ze strany serveru.

Komunikace standartně probíhá cyklicky v definovaném čase jednoho cyklu. Tento čas musí být pro řízení pohonů dostatečně krátký, což může být u ADS komunikace potencionální problém. Event-based komunikace je možná pouze pro čtení dat ze serveru a probíhá na základě předem definované události. Implementace do Simulink modelu se provede pomocí bloků, které jsou při instalaci TwinCAT 3 Interface for MATLAB/Simulink automaticky nahrány do knihovny. Beckhoff TwinCAT následně přistupuje k datům pomocí vytvořeného TcCOM objektu.

### 3.2.4 TwinCAT 3 a LabVIEW

Komunikace mezi vývojovým prostředím NI LabVIEW a Beckhoff TwinCAT 3 runtime je stejně jako u Matlabu zprostředkována přes ADS komunikaci. K dispozici je modul TwinCAT 3 Interface for LabVIEW, který automaticky nainstaluje všechny komunikační ADS klient bloky do LabVIEW, ve kterém se využívají obdobně jako u Simulinku. Komunikace může probíhat cyklicky nebo event-based, synchronně nebo asynchronně. V praxi se uplatňují při paralelním běhu TwinCATu a LabVIEW dva přístupy. Minimalistická konfigurace využívá TwinCAT pouze jako ovladač pro hardwarové sběrnice, fyzické vstupy a výstupy jsou přivedené do LabVIEW, ve kterém jsou odbavovány programem. Druhá konfigurace odbavuje celý program v TwinCATu a LabVIEW využívá pouze jako uživatelské rozhraní HMI.

V době psaní této práce společnost Beckhoff představila chystanou novinku s názvem TwinCAT 3 Beckhoff I/O VIs for LabVIEW, která by měla umožňovat snadné propojení hardwaru od National Instruments s hardwarem Beckhoff. Obsahuje programové bloky (VI), pomocí kterých je možné přímo v LabVIEW konfigurovat nastavení externích Beckhoff periferií a terminálů. Lze zde nastavit například filtry vstupů, měřicí rozsahy, vzorkovací frekvence a podobně. Program se v LabVIEW pomocí LabVIEW Real-Time modulu převede na aplikaci, která se nahraje do zařízení od National Instruments. K tomuto zařízení se poté připojí periferie od Beckhoffu. Do této chvíle bylo možné propojovat EtherCAT zařízení s hardwarem od NI pouze pomocí předpřipravených konfiguračních XML souborů. Pokud XML soubory nebyly dostupné, bylo je nutné vytvořit. TwinCAT 3 Beckhoff I/O VIs for LabVIEW by měl konfiguraci hardwaru výrazně zjednodušit.

### 3.2.5 Propojení nástroje REXYGEN s PLC a Simulinkem

Jednou z možností programování v REXYGEN Studiu je jazyk Function Block Diagram podle normy IEC 61131-3. Kvůli tomu je možná velice snadná implementace kódu vytvořeného v tomto vývojovém prostředí do průmyslového hardwaru jiných výrobců. Pro komunikaci mezi PLC firmy SIEMENS a run-time RexCore nabízí Rex Controls ovladač s názvem S7Drv module. RexCore může operovat v režimu klient, kdy emuluje PG (programming device – např. notebook), nebo v režimu server, kdy emuluje jiné SIEMENS PLC. Komunikace probíhá synchronně nebo asynchronně. [15]

Algoritmy lze též vyvíjet pomocí Matlabu a Simulinku, a to s využitím knihovny RexLib, která se nahraje do Simulinku. V něm se vytvoří řídicí algoritmus, při vývoji lze využívat Matlab Toolboxy, například pro ladění regulátoru nebo identifikaci parametrů. Výsledný algoritmus obsahující pouze prvky z knihovny RexLib je exportován a spuštěn v zařízení s RexCore run-time. [16] Podobná knihovna též existuje pro vývojové prostředí LabVIEW.

### 3.2.6 LabVIEW Model Interface Toolkit

Vzhledem k tomu, že na externí real-time zařízení propojená se Simulinkem má monopol společnost Speedgoat, může se v některých případech nabízet vytvoření programu v Simulinku s využitím pokročilých toolboxů, jako například Control System Toolbox, a následná implementace algoritmu v konkurenčním vývojovém prostředí, které podporuje rozmanitější, leckdy i levnější nabídku real-time zařízení. Právě k tomu slouží softwarový add-on pro LabVIEW s názvem LabVIEW Model Interface Toolkit, umožňující implementaci modelů ze Simulinku nejen do vývojového prostředí LabVIEW, ale i do LabVIEW Real-Time Module. Díky tomu je možné provést snadnou implementaci modelu do real-time zařízení, například do průmyslového zařízení Compact RIO. Modul také umožňuje zavedení C/C++ kódu. Další užitečnou funkcí je propojení LabVIEW a Simulink modelů a možnost jejich kosimulace. Uplatnění může najít například při MiL simulacích (Model in the Loop). V Simulinku je s využitím add-onu Simscape vytvořen model mechanické struktury stroje a v LabVIEW běží řídicí algoritmus, kterým je stroj ovládán. Komunikaci mechanického virtuálního modelu s řídicím systémem zajišťuje LabVIEW Model Interface Toolkit.

### 3.2.7 Beckhoff Motion Control a Drive Manager

Vývojové prostředí TwinCAT 3 nabízí řadu modulů přímo určených pro ovládání pohonů. Moduly jsou součástí skupiny TwinCAT 3 Motion Control a každý z nich je provozován pod vlastní licencí. Základní varianta s názvem NC PTP umožňuje pohyb point-to-point bez nastavení trajektorie. Pohon se rozjede na požadovanou rychlost s požadovaným zrychlením, polohuje k zadanému bodu a s požadovaným zrychlením do něj opět zpomalí. Je možné zadávat hodnoty maximálních jerků. Pro komunikaci s reálným pohonem je vytvořen v řídicím systému abstraktní objekt „axis,“ který je propojen s PLC a zprostředkovává pokyny pro pohon. Tento objekt je stejný pro všechny výrobce a typy pohonů – elektrická a hydraulická serva, krokové motory atd., čímž se zjednodušuje praktická implementace os při psaní programu. [17]

Pro komplikovanější pohyby jsou ve skupině Motion Control k dispozici další moduly. TwinCAT NC I (numerical control interpolation) slouží k interpolování pohybu mezi řadou zadaných bodů, a to nejen pro jednu osu, ale i pro prostorové pohyby. Na tento modul přímo navazuje vrcholný TwinCAT 3 CNC pro interpolování až 32 os, s podporou transformace souřadných systémů pro optimální průběh dráhy stroje. Zadávání dráhy a příkazů ve formě G-kódu dle normy DIN 66025 je také podporováno.

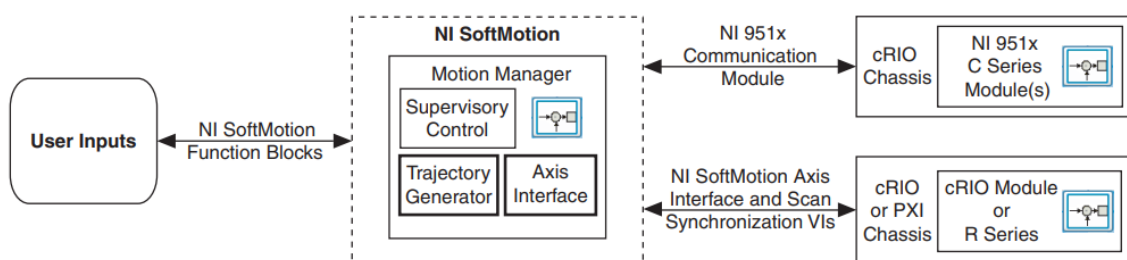
Pohyby několika strojních os mohou být svázány, Beckhoff toto propojení nazývá vždy „master-slave,“ ačkoliv se může jednat o různé druhy propojení. Axis coupling v rámci knihovny Tc2\_MC2 umožňuje vložit mezi osy virtuální převod, převodový poměr se navíc může dynamicky měnit při běhu programu. Modul NC Flying saw synchronizuje polohu a rychlost jedné osy s druhou osou, která je již v pohybu. Jméno je odvozeno od pily, která synchronizuje svou rychlost s projíždějícím polotovarem, provede řez a vrátí se do původní polohy, aniž by se polotovar musel zastavit. Další možností propojení os je NC Camming. Jedná se o virtuální vačku, na základě pohybu master osy se přiřazují slave ose požadavky na polohu. Hodnoty mohou být brány z připravené tabulky, vypočítány pomocí matematického výrazu nebo načítány externě.

Pokud je pohon od společnosti Beckhoff, jeho propojení s řídicím systémem je možné snadno zajistit softwarem TC3 Drive Manager 2 (dříve Drive Manager 1, zpětná kompatibilita je zajištěna). Jedná se o nezávislý projekt ve vývojovém prostředí, který je schopný automaticky naskenovat pohony nebo je manuálně vybrat ze seznamu.

V rámci TwinCAT 3 Engineering je k dispozici software pro výběr a dimenzování pohonů s názvem TwinCAT 3 Motion Designer. Software podporuje běžné mechanické převody, jmenovitě hřeben – pastorek, kuličkový šroub, klikovou hřídel a další. Jako vstupní požadavky se zadají mechanické vlastnosti soustavy a požadavky na dráhu a dynamiku pohybů. Program následně provede optimalizační výpočet, v rámci kterého navrhne vhodné kombinace pohonů a k nim příslušné převodové poměry, napájecí zdroje a příslušenství, např. brzdný rezistor o dostatečné kapacitě. Samozřejmostí je kompatibilita vybraných komponent, jejichž seznam je generován v rámci reportu. Dimenzování pohonů je možné provést také pro víceosé stroje.

### 3.2.8 NI SoftMotion

National Instruments podobně jako Beckhoff nabízí modul pro ovládání pohonů s názvem NI SoftMotion. [18] Pro komunikaci programu s modulem jsou předpřipravené funkční bloky (function blocks, viz Obrázek 3-9). V NI SoftMotion poté probíhá odbavování os. Část Supervisory Control obsahuje hlavní smyčku pro řízení pohybů. Zpracovává uživatelské požadavky na polohu a zprostředkovává je dále do interpolátoru. Zároveň kontroluje příslušné vstupy/výstupy, například koncové spínače pro referování os nebo senzory pro indikaci chyby osy.



Obrázek 3-9 Propojení modulu NI SoftMotion [18]

Uvnitř Trajectory Generator běží interpolátor, který vytváří setpointy pro regulační smyčku na základě požadované trajektorie. Pro plynulejší generování trajektorie je k dispozici Spline Engine, který interpoluje dráhy jako kubický spline. Pro krokové motory je možné použít Stepper Generator, jenž bere data ze spline křivky a převádí je na kroky motoru.

Uvnitř části Control Loop operuje regulační smyčka pohonu, v případě servopohonů obvykle kaskádní regulace. Vstupem je odměřování polohy, rychlost je poté dopočítána. Fyzické osy pohonu jsou uvnitř NI SoftMotion konfigurovány jako abstraktní objekty s názvem Axis Interface. NI SoftMotion může být implementován například do systému Compact RIO (cRIO), ke kterému mohou být připojeny moduly NI 951x od National Instruments, nebo jiné kompatibilní pohony od dalších výrobců. [19] Mezi těmito osami a řízením v NI SoftMotion poté probíhá nízkourovňová komunikace zprostředkovaná pomocí objektu Axis Interface.

### 3.2.9 FPGA – programovatelné hradlové pole

Regulační smyčka a další části programu mohou být provozovány ve vysokých takttech na programovatelném hradlovém poli – FPGA. Jedná se o logický obvod, který obsahuje jednoduché logické prvky (tranzistory a hradla) a programovatelné propojovací vodiče (přepínače s pamětí). Výsledný logický obvod tedy může být libovolně naprogramován koncovým uživatelem. Oproti tomu u standartního integrovaného polovodičového čipu návrhář sestaví logický obvod přímo u výrobce. Na základě tohoto návrhu se začnou na výrobní lince vyrábět čipy, u kterých již nelze měnit propojení logických prvků. Hlavním důvodem pro použití FPGA čipu oproti procesoru běžné architektury je možnost provádění několika logických operací paralelně v jednom taktu čipu. Tím se může razantně zvýšit rychlost zpracovávání signálu v reálném čase. Klasický procesor v jednom taktu provádí pouze jednu logickou operaci. Nevýhodou FPGA je vyšší komplikovanost, a tedy i cena. Tyto obvody se nevyplátí pro série nad 10 000 kusů. [20]

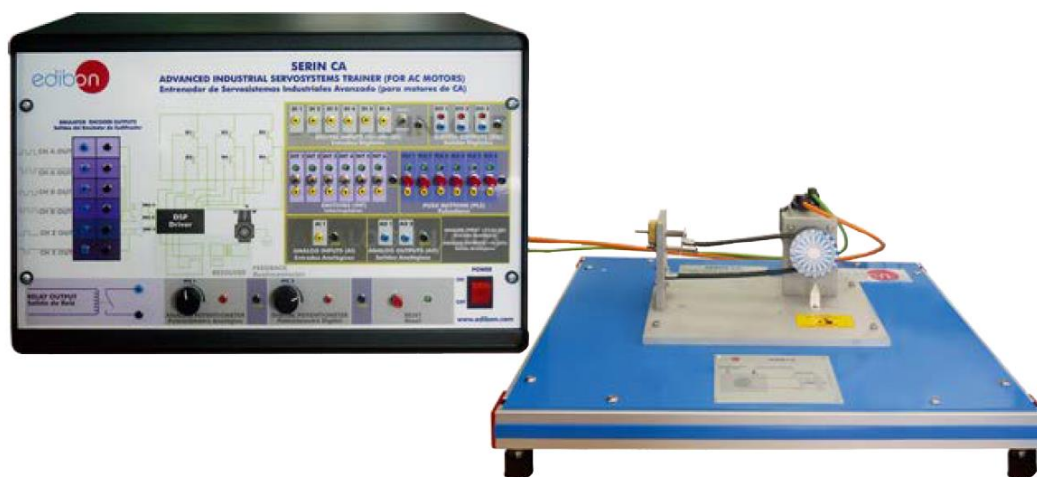
Programovatelná hradlová pole se běžně programují jazykem HDL. National Instruments nabízí LabVIEW FPGA Module, s jehož pomocí je možné programovat obvody v grafickém programovacím jazyce LabVIEW. Vytvořený kód je možný implementovat například do systémů CompactRIO. [21] Mathworks nabízí HDL Coder, jenž převede program vytvořený v Simulinku nebo Simscape na HDL kód pro FPGA obvody, podobně jako PLC Coder. Kód je možné optimalizovat pro rychlost nebo propustnost. Generovaný kód je strukturovaný, komentovaný a kompatibilní s běžnými výrobci FPGA obvodů, jako jsou Xilinx a Intel. [22]

### 3.3 Rešerše standů se zaměřením na řízení rotačních pohonů

V tomto úseku práce jsou prezentovány komerčně dostupné standy používané pro výuku. Jsou zde také uvedeny pracoviště z odborných publikací, určená pro laboratorní testování, měření a verifikování řídicích algoritmů. Jedná se o různé kombinace pohonů, řídicích systémů a konstrukcí, které mohou sloužit jako inspirace při návrhu vlastního standu.

#### 3.3.1 Edibon SERIN/CA

Edibon je španělská společnost, zabývající se prodejem výukových zařízení pro technické školy. V tomto oboru má mnohaletou zkušenost a její produkty jsou využívány po celém světě. Pro řízení servopohonů nabízí středně velký stand SERIN/CA s hmotností 40 kg (Obrázek 3-10). Set tvoří řídicí hardware Control Interface Box, ke kterému je připojený střídavý servomotor a počítač. Součástí je také mechanická brzda působící na hřídel motoru, která může simulovat zátěž a vnášet chybu do řídicího algoritmu. Úhel natočení je odečítán resolverem umístěným na servomotoru. Komunikace s počítačem probíhá přes rozhraní RS232 v reálném čase. Control Interface Box obsahuje trojici digitálních výstupů. V základní konfiguraci první výstup indikuje, jestli se motor nachází v chybovém stavu. Druhý výstup značí zapnutí/vypnutí brzdy. Všechny tři výstupy mohou být libovolně naprogramovány. Box disponuje čtyřmi analogovými výstupy, jedním analogovým vstupem, šesti tlačítky a šesti přepínači a emulátorem enkodéru. Emulátor přepočítává výstup resolveru na TTL signál napodobující pulzy enkodéru. Samotný servomotor dosahuje výkonu 0,7 kW, maximálních otáček 4 200 ot./min a krouticího momentu 7,2 Nm.



Obrázek 3-10 -Výuková stanice Edibon SERIN/CA [23]



Pro řízení a zobrazování aktuálních a naměřených hodnot využívá Edibon vlastní řídicí software Real Time Control System (RTC). Software provede uživatele detailní konfigurací, ve které nejprve nastaví proměnné pro digitální vstupy a výstupy. Zde může volit z řady předpřipravených funkcí. Následuje obdobné nastavení analogových vstupů a výstupů. Další záložku tvoří již nastavení parametrů kaskádní regulační smyčky – proporcionálních zesílení a integračních konstant. Pro řízení je k dispozici řada operačních módu a příslušných zdrojů signálu. V polohovém módu je vstupem požadavek polohy a může se jednat o předpřipravený pohyb nebo o signál přivedený na digitální vstup. V rychlostním módu může být pevně nastavena požadovaná rychlost nebo může být měněna na základě digitálního vstupu, například z potenciometru. V momentovém módu je regulován kroutící moment na motoru a vstupní signál může být rovněž předdefinovaný nebo analogový externí.

Pohybové sekvence mohou být ukládány pod jednotlivá hardwarová tlačítka na řídicím boxu. Samozřejmostí je nastavení maximálních otáček a zrychlení. Po nastavení všech parametrů následuje spuštění stanice. Pro sledování aktuálních hodnot je přiložený software RTC vybavený osciloskopem, ve kterém se zobrazuje stav všech veličin, ať už se jedná o požadované hodnoty rychlosti či polohy, nebo o aktuální rychlost, vstupy a výstupy, napětí nebo hodnoty teplot z teplotního sensorů motoru. V online režimu může obsluha za běhu měnit parametry regulační smyčky a v osciloskopu sledovat změny v chování soustavy.

Pro výuku poskytuje Edibon ke svým produktům software typu SCADA, podporující až 30 studentů pracujících na jedné hardwarové stanici, v tomto případě SERIN/CA. Pedagog nejprve vytvoří úlohy v EDIBON Classroom Manager, které rozešle studentům do jejich PC se softwarem EDIBON Student Labsoft. Studenti mají vlastní účty, kterými se do softwaru přihlásí a začnou úlohy plnit. Učitel z Classroom Manageru vidí aktuální stav práce studentů. Studenti mohou upravovat parametry stanice SERIN/CA v rámci softwaru Student Labsoft a pak je následně na fyzickou stanici nahrát a verifikovat si nastavení regulátoru. Učitel může také provádět experimenty na výukovém standu a studenti poté sledují aktuální parametry nastavení, osciloskop a další údaje přímo z vlastních pracovních stanic. [23]



### 3.3.2 Googoltech GSMT2014

Firma Googoltech nabízí testovací stand pro výuku pohonů, automatizace a základních principů regulace. Sestavu (Obrázek 3-11) tvoří hliníková základna, na které je ve dvojici domků uložená poddajná hřídel. Na hřídeli jsou umístěny setrvačnosti a oba konce pohání přes ozubené řemeny DC servomotory o výkonu 70 W. Na samotném konci hřídele se nachází dvojice rotačních enkodérů s rozlišením 1 000 pulzů na otáčku.

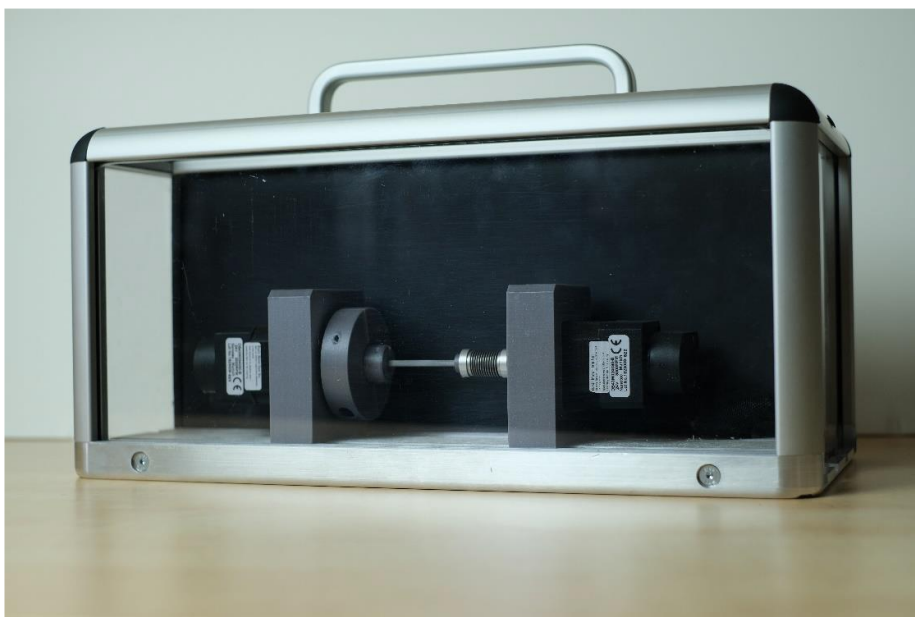


*Obrázek 3-11 - GSMT2014 [24]*

O řízení motorů se stará PCI karta GT-400 vyvinutá přímo společností Googol Technology Ltd. Karta zvládne řídit až 4 osy s krokovými nebo servo motory, disponuje čtyřmi vstupy rotačních enkodérů, DSP a FPGA technologií. Digitální signálový procesor DSP je mikroprocesor optimalizovaný pro zpracování digitálních signálů, uzpůsobený pro velký průtok dat. V jednom taktu je na rozdíl od klasických procesorů schopný zvládat více operací. Větší množství dat může být zpracováno pomocí programovatelného hradlového pole FPGA. [24] Karta GT-400 je zapojená v počítači a komunikuje s programem Matlab SLDRT, který umožňuje řízení experimentů v reálném čase. Googoltech nabízí ke standu celou řadu experimentů, jako například vyšetřování vlastních frekvencí systému, přenosové funkce, kalibraci PID regulátoru, stavovou zpětnou vazbu a různé principy řízení pohonů. [25]

### 3.3.3 Flexible two-mass motion system

Flexible two-mass motion system je výukový stand vyvinutý na Západočeské univerzitě v Plzni Martinem Goubajem a Lukášem Bláhou. [26] Se vzrůstajícími požadavky na přesnost a rychlost regulačních systémů servopohonů se také zvyšují nároky na studenty technických univerzit. Důraz je kladen především na pochopení interakce mezi mechanickou strukturou a řídicím algoritmem. Pro potřeby výuky bylo na ZČU vyvinuto kompaktní testovací pracoviště, složené z dvojice servomotorů, propojených poddajnou hřídelí, s předem vypočtenou torzní tuhostí tak, aby měla soustava požadované vlastní frekvence (Obrázek 3-12). Korekce nesouosostí servomotorů kompenzuje pružná spojka mezi hřídelí a jedním z motorů. Velikost jedné ze setrvačností je možné upravovat přidavným setrvačником, který je upevněný přímo k hřídeli jednoho ze servomotorů. Není tím tedy ovlivněna tuhost soustavy. Servomotory jsou namontovány na drážkované hliníkové desce pomocí 3D tištěných držáků z PLA. Celý pracovní prostor kryje konstrukce z profilů 20x20 mm, stejně tak i řídicí elektroniku skrytou v zadní v části standu. Kontakt rotujících částí s obsluhou brání polykarbonátové desky vsunuté do profilů. Hmotnost celé soustavy činí méně než 8 kg.



*Obrázek 3-12 - Flexible two-mass motion systém [26]*

Samotné pohony jsou stejnosměrné bezkartáčové (BLDC) servomotory Nanotec DB43M024030 s výkonem 53 W, jmenovitými otáčkami 3 000 ot./min. a krouticím momentem 17 Ncm. Úhel natočení je odměřován optickým inkrementálním enkodérem s rozlišením 20 000 CPR (Counts Per Revolution – počet cyklů na otáčku). Napájení motorů zajišťují frekvenční měniče Maxon MAXPOS 50/5 určené pro motory do výkonu 250 W. Přes měnič může být také uzavíráno zpětnovazební řízení, proudová smyčka dosahuje taktu až 10  $\mu$ s (100 kHz), rychlostní smyčka 100  $\mu$ s (10 kHz) a polohová smyčka rovněž 100  $\mu$ s (10 kHz). K dispozici je také rychlostní feedforward. Komunikace mezi měničem a řídicím systémem probíhá přes průmyslovou komunikační sběrnici EtherCAT, řídicí systém zde vystupuje jako Master, měniče Maxon jako Slave. Alternativní možností konfigurace je využití USB 2.0/3.0 portu. Měnič disponuje funkcemi pro zadávání požadavků na pohyb (režim point to point) a na profil rychlosti. [27]

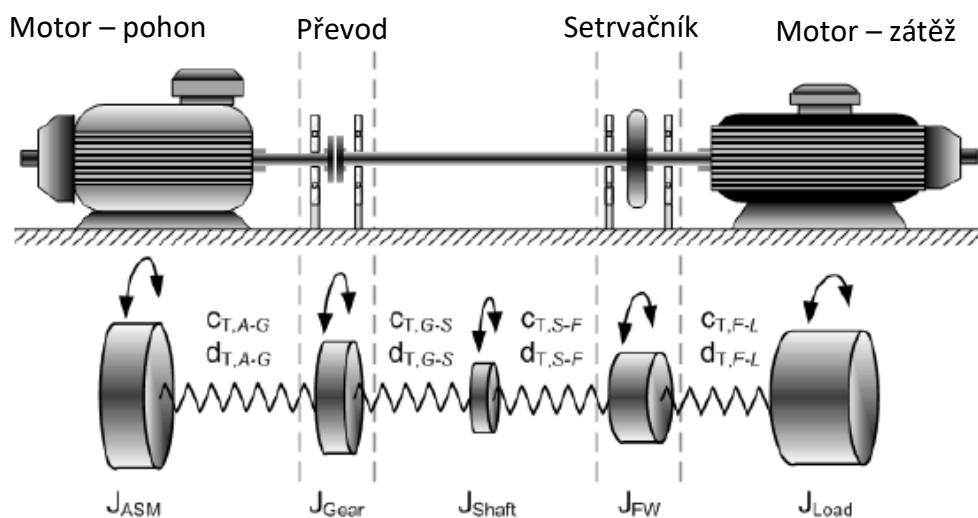
V případě standu z publikace [26] bylo pro řízení použito Raspberry Pi 4 ve výkonnější variantě se čtyřjádrovým procesorem o taktu 1,5 GHz a operační paměti 8 GB LPDDR4. Komunikace s měničem probíhá přes EtherCAT. Raspberry Pi je vybaveno sběrnici Monarco HAT I/O, která zařízení rozšiřuje o čtyři digitální vstupy a výstupy a další dva analogové vstupy a výstupy. Dále obsahuje konektor RS-485, devět signalizačních LED diod, to vše v kovové krabici určené pro montáž na DIN lištu. [28] Raspberry Pi běží na real-time operačním systému RexCore, programování probíhá ve vývojovém prostředí REXYGEN. Proudová regulační smyčka je uzavřena přes frekvenční měnič, filtrovaná zpětnovazební smyčka rychlosti a zpětnovazební smyčka polohy jsou uzavřeny skrze Raspberry Pi. To také generuje trajektorii a rychlostní feedforward.

Autoři v publikaci popisují sestavený matematický model mechanické soustavy a regulační úlohy, které jsou na standu při výuce prováděny. Jde především o ladění regulační smyčky, nejprve manuální změnou parametrů regulace, poté pomocí různých ladících metod, aplikací filtrů apod. Ladění regulátoru nemusí probíhat výhradně na hardwarovém standu, celá soustava může být simulována v Simulinku s řídicím algoritmem vytvořeným pomocí knihovny RexLib pro Matlab. Při ladění lze využít toolboxu Simulinku, jako například Control System Designer, které usnadňují naladění a přenosovými funkcemi vizualizují vliv regulačních parametrů na kvalitu regulace.

Každý student může naladit simulovanou soustavu v Simulinku na vlastním počítači a poté experimentálně ověřit na výukovém standu. Soustava může být v režimu, kdy je poháněn jeden servomotor a druhý působí pouze jako setrvačnost, s polohovou regulační smyčkou uzavřenou z hnaného nebo hnacího motoru. Dále je možné využívat druhý motor jako zdroj chyby, simulovat cogging, kolísání krouticího momentu vlivem elektronické komutace nebo nevývahu rotoru.

### 3.3.4 Testovací stand z publikace Hoffmanna a kol.

Tento testovací stand využil Hoffmann a kol. v rámci publikace Model Based Predictive Speed Control of a Drive System with Torsional Loads – A Practical Approach. [29] V této práci autoři navrhli prediktivní řízení rychlosti servomotoru založené na matematickém modelu. Algoritmus je určen pro poddajné pohony zatížené torzním namáháním, jako jsou například dlouhé a tenké hřídele. Torzní oscilace mohou působit negativně na životnost pohonu, způsobovat únavová poškození, ale především snižují pásmo propustnosti rychlostní smyčky kaskádní regulace.



Obrázek 3-13 – Schéma testovacího standu a torzní model soustavy [29]

Sestavu, na které byl řídicí algoritmus verifikován, tvoří dvojice elektromotorů, propojených převodem a poddajnou hřídelí. Hřídel je uložena v ložiskových domkách, na jedné straně je na ní navíc připevněn setrvačnick. Autoři modelovali soustavu jako pětihmotový systém – dva rotory motorů, setrvačnost hřídele, setrvačnost převodů a samotný setrvačnick (Obrázek 3-13).

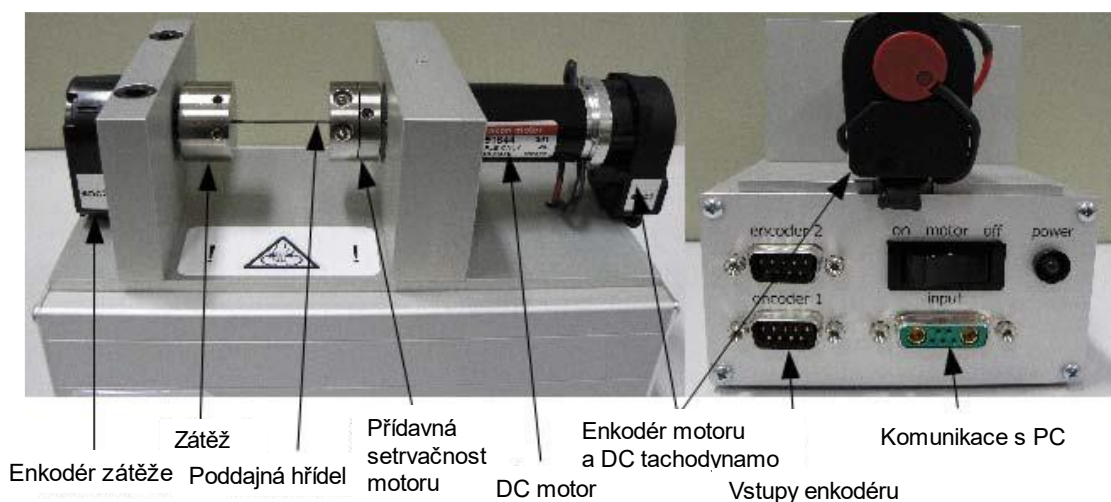
Setrvačnosti jsou propojené příslušnými tuhostmi a tlumením. Jako hnací motor byl zvolen střídavý synchronní servomotor s permanentními magnety o výkonu 6,4 kW, krouticím momentu 24,6 Nm a jmenovitých otáčkách 2 490 ot./min. Motor je napájen frekvenčním měničem. Druhý motor tvoří zátěž pro servomotor, proto byl zvolen levnější a mechanicky odolnější asynchronní motor klecové konstrukce (squirrel cage). Výkon dosahuje hodnoty 5,5 kW, krouticí moment 36 Nm a jmenovité otáčky 1 455 ot./min. K asynchronnímu motoru je připevněn enkodér pro odečítání polohy a rychlosti. Za převody může být též přidán snímač krouticího momentu.

Řídicí algoritmus s prediktivním řízením rychlosti byl vytvořen v Matlab Simulinku, ze kterého byl vygenerován C/C++ kód, který byl následně implementován prostřednictvím řídicí desky DS 1103 od německé společnosti dSPACE. Testovací stand se v praxi osvědčil, navrhnutý algoritmus se podařilo s jeho využitím verifikovat, navíc bylo možné změřit torzní namáhání hřídele a porovnat přínos prediktivního řízení otáček servomotoru oproti běžnému PI regulátoru. Stand svojí konstrukcí a velikostí odpovídá běžným pohonům v praxi, proto lze předpokládat podobné výsledky i při aplikaci navržených algoritmů. Daní za to jsou rozměry a hmotnost testovací sestavy.

### 3.3.5 Motion Control Demonstrator

Saey a kol. z belgické Katholieke Universiteit Leuven prezentovali výuku torzně namáhaných pohonů v práci s názvem Drive systems with torsional load: versatile low-cost educational laboratory set-up. [30] K výuce využívají stand s názvem Motion Control Demonstrator (MCD) vyvinutý společností IME, MathWorks a pracovníky z německé Eindhoven University of Technology (Obrázek 3-14). MCD je tvořen hliníkovým boxem pro regulační elektroniku, na kterém jsou umístěny dva hliníkové bloky. První z nich slouží pro upevnění stejnosměrného motoru, rotačního inkrementálního enkodéru a DC tachodynamu. Hřídel motoru je osazena přídatnou setrvačností. Ve druhém hliníkovém bloku je uložena další setrvačnost a rotační enkodér. Setrvačnosti jsou propojeny poddajnou hřídelí a jejich velikosti lze měnit ocelovými kroužky, které se navlíknou na disk a upevní se svěrným spojením pomocí stavěcího šroubu s vnitřním šestihranem.

O řízení MCD se stará stolní počítač se starším real-time operačním systémem xPC Target od MathWorks. Jedná se o předchůdce softwaru Simulink Desktop Real Time a Simulink Real Time, jenž je provozován v stand-alone režimu na běžném desktopovém hardwaru. [31] Pro připojení digitálních a analogových vstupů a výstupů byl počítač osazen PCI-E kartou od Humusoftu.



Obrázek 3-14 - Motion Control Demonstrator [30]

## 4 Shrnutí rešerše

Rešeršní část práce se v úvodu zabývala porovnáním vhodných řídicích systémů pro servopohony. Ve druhé sekci následoval výčet komerčně vyráběných a laboratorně používaných pracovišť se zaměřením na řízení rotačních servopohonů. Jako zástupce průmyslové řídicí platformy byl v rešerši uveden TwinCAT od společnosti Beckhoff. Mezi přednosti tohoto systému patří možnost provozu na počítači běžné architektury, snadné propojení s pohony od stejnojmenného výrobce a dobrá konektivita s ostatními vývojovými prostředími. IDE samotného TwinCATu je zdarma a pro běhové prostředí je možné využívat sedmidenní trial licence.

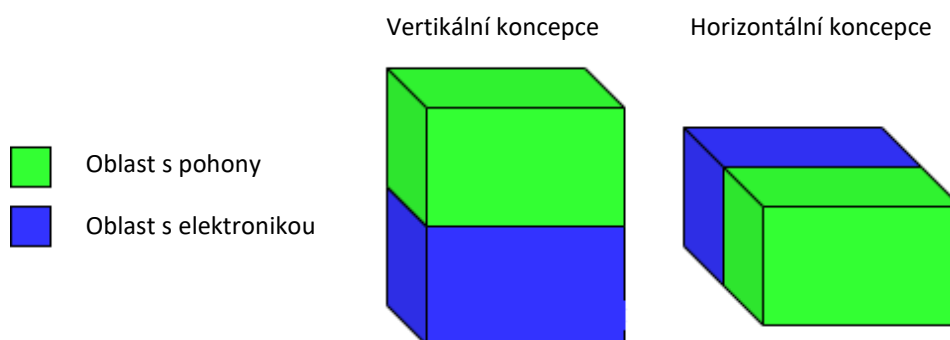
Českou alternativu k řízení od Beckhoffu tvoří nástroj REXYGEN s využitím průmyslového PLC. Tento software úspěšně použili autoři publikace [26] při realizaci dvoumotového standu. Výhodou je relativně nízká cena, proti mluví nižší rozšíření v průmyslu a hůře zpracovaná dokumentace. Posledním řídicím softwarem, který by připadal v úvahu, je LabVIEW společnosti National Instruments v kombinaci s Real-Time modulem. Jedná se o dražší řešení, které je možné provozovat pouze na hardwaru společnosti NI. Mezi výhody patří dobrá dokumentace a rozšíření především v oblasti sběru a vyhodnocování dat. Pro ovládání pohonů je k dispozici modul NI SoftMotion podporující serva a měniče různých výrobců. Nejlepší vývojové prostředí pro tvorbu nových řídicích algoritmů nabízí Matlab Simulink. Disponuje toolboxy přímo navrženými pro ladění regulace a vývoj virtuálních modelů. Jedná se ale také o nejdražší variantu. Běh v plném výkonu je možný pouze na dedikovaném zařízení od společnosti Speedgoat. Ostatní výrobci ovšem umožňují snadné propojení jejich softwaru se Simulinkem, ať už se jedná o TC3 Target for Simulink od Beckhoffu, LabVIEW Model Interface Toolkit od National Instruments nebo RexLib od REX Controls. Nabízí se tedy možnost vyvinout řídicí algoritmus v Simulinku a následně ho implementovat do vhodnějšího hardwaru.

Z rešerše standů vyplynuly požadavky na samotné pracoviště. Mělo by se jednat o kompaktní, snadno přenositelný celek robustní konstrukce. Výhodou je možnost simulovat mechanickou soustavu a regulaci na odděleném počítači a následně přenést parametry regulační smyčky do fyzického standu. Zařízení by nemělo být příliš drahé, aby bylo možné jeho nasazení při výuce ve větším počtu.



## 5 Návrh koncepčních variant pracoviště

Při návrhu standu pro výuku servopohonů lze volit řadu konstrukčních a designových variant, řídicích systémů, pohonů a následně stanovit optimální rozložení a uspořádání komponent. Mechanická a elektronická část standu může být umístěna v jednom společném prostoru, případně může být oddělena do dvou uzavřených oblastí, umístěných vedle sebe (horizontální koncepce) nebo nad sebou (vertikální koncepce) viz Obrázek 5-1:



Obrázek 5-1 Možnosti rozmístění komponent

Řídicí systémy lze kombinovat s různými výrobci pohonů, z čehož vyplývají rozdílné požadavky jak na samotný řídicí software, tak na náročnost implementace. V některých případech jsou regulační smyčky součástí měničů, jindy jsou výhradně v řídicím počítači a do měničů je posílán pouze požadavek na proud. Na trhu jsou také kombinace těchto variant. Dostupné velikostní řady pohonů mají vliv i na samotnou mechanickou část, ovlivňují velikosti setrvačných hmot a požadavky na torzní tuhost soustavy, čímž dochází ke změně modálních vlastností a tím i velikosti celého pracoviště. Vlastní frekvence systému by měla být dostatečně nízká tak, aby se při ladění pohonů projevovaly primárně jejich mechanické vlastnosti podobně jako u reálného stroje. V případě vyšších vlastních frekvencí by do ladění mohl nepříznivě vstupovat takt regulace, komunikace řídicího systému s měniči a také výsledné dopravní zpoždění signálu.

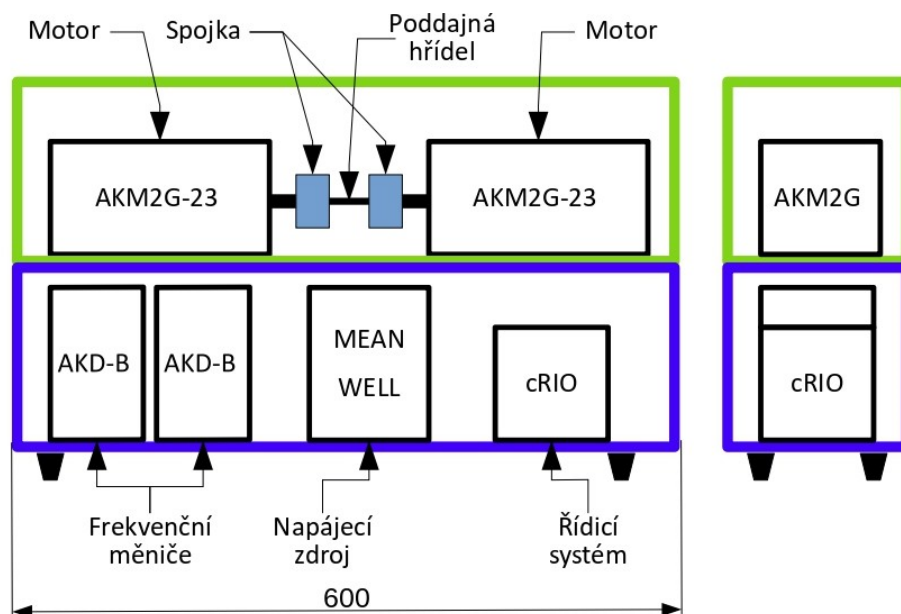
Pro volbu optimální konfigurace standu byly vytvořeny tři varianty, ze kterých byla vybrána výsledná koncepce pracoviště. Každá z variant je založena na jiném řídicím systému s pohony rozdílných výrobců. Zároveň bylo každé variantě přiřazeno odlišné rozmístění elektroniky, pohonů a mechanických komponent tak, aby bylo možné jednotlivé varianty efektivně porovnat a vybrat vhodné výsledné řešení.



## 5.1 Varianta 1 – LabVIEW s pohony Kollmorgen

První navrhovaná varianta je založená na platformě LabVIEW společnosti National Instruments s využitím modulu Real-Time pro možnost řízení v reálném čase. Pro implementaci řídicího algoritmu byl zvolen hardware CompactRIO (cRIO). Mezi hlavní důvody pro jeho volbu patří hojně rozšíření cRIO v průmyslu, vysoký výpočetní výkon napříč dostupnými variantami, rozumné ceny v rámci nabídky NI a možnost využití integrovaného FPGA obvodu pro řídicí algoritmy s vysokým taktům až 40 MHz. Systém může běžet samostatně bez připojeného počítače s vývojovým prostředím, případně může být propojený USB/Ethernetovým kabelem a využíván jako HMI.

Pohony byly vybrány od americké společnosti Kollmorgen, především kvůli kompatibilitě s hardwarem National Instruments a dostupnosti v České republice, kde má Kollmorgen zastoupení. Motor byl zvolen AKM2G-23 s výkonem téměř 1 kW, krouticím momentem 1,5 Nm a možností napájení 48 V. Řízení motoru zajišťuje měnič Kollmorgen AKD-B s podporou EtherCAT komunikace, jenž slouží pro propojení s cRIO. O obsluhu pohonů se stará modul NI SoftMotion s integrovaným interpolátorem a generátorem setpointů. Polohová a rychlostní regulační smyčka může být uzavřena jak v měniči, tak v řídicím systému. Komponenty jsou zde rozmístěny nad sebou v koncepci nastojato. Měniče, napájecí zdroje a řídicí systém jsou umístěny ve spodním patře standu. Ve vrchním patře se nachází mechanická část, kterou tvoří dvojice motorů propojených poddajnou hřídelí (Obrázek 5-2):



Obrázek 5-2 – Varianta 1

Vlastní frekvenci zde určuje téměř výhradně průměr a délka poddajné hřídele, protože setrvačnost spojek je zanedbatelná a setrvačnost rotorových částí motorů lze měnit pouze změnou motoru. Na druhou stranu se kvůli této jednoduchosti chování sestavy velmi blíží dvouhmotovému systému.

Implementace řídicích algoritmů z Matlabu je možná s využitím LabVIEW Model Interface Toolkit. S využitím pokročilých toolboxů lze vytvářet například stavové zpětné vazby a další pokročilé formy řízení. Vytvořený program se zkompiluje a nasadí přímo na cRIO. Druhou možností je propojení cRIO s počítačem, na kterém je nainstalováno LabVIEW a Matlab. Vzájemná komunikace poté může probíhat v reálném čase, což je vhodné například pro analýzu signálů v případě tvorby Bode diagramů přenosů.

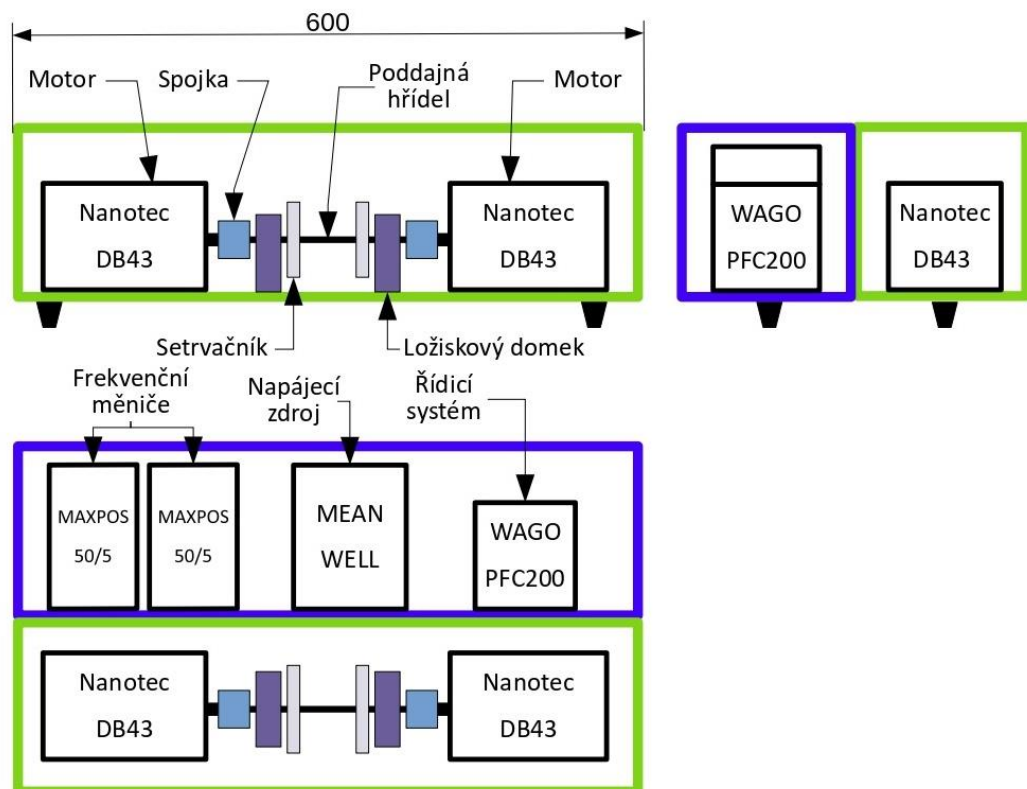
## 5.2 Varianta 2 – REXYGEN s pohony Nanotec a Maxon

Druhá varianta je postavena na řídicím systému REXYGEN, který je nasazen na WAGO PFC200. Jedná se o průmyslové PLC, běžící na procesoru Cortex A8 a real-time Linuxovém jádře s podporou RTOS RexCore. Konektivitu zajišťují vstupní a výstupní karty, kterých lze do jednoho procesorového modulu zapojit až 32. Komunikace s počítačem probíhá pomocí ethernetového kabelu a stejně jako u předešlé varianty může stand operovat nezávisle, nebo v kooperaci s počítačem.

Kvůli absenci FPGA obvodu a celkově nižšímu výkonu chipsetu se nedoporučují takty komunikace vyšší než 5 ms (200 Hz). Tento fakt výrazně omezuje možnosti uzavírání polohové a rychlostní smyčky z řídicího systému. Smyčky obvykle běží v nižším taktu 100 ms, respektive 10 ms, do hry ovšem vstupuje dopravní zpoždění, které by v tomto případě mohlo být příliš velké pro plynulý chod.

Pohony v této variantě tvoří bezkartáčové stejnosměrné šestipólové motory Nanotec DB43 o výkonu 0,138 kW a krouticím momentu 0,44 Nm. O řízení motorů se stará měnič MAXPOS 50/5, podporující motory do výkonu 0,25 kW. Umožňuje zapojení inkrementálních i absolutních enkodérů a uzavírání proudové, rychlostní a polohové smyčky v měniči v taktech 100 kHz pro proudovou a shodně 10 kHz pro rychlostní a polohovou smyčku. Komunikace s řídicím PLC probíhá přes EtherCAT sběrnici, kterou WAGO PFC200 na rozdíl od svého předchůdce PFC100 podporuje. Samozřejmostí je možnost implementace feed-forwardů pro další zvýšení propustného pásma regulace.

Rozložení komponent je u této varianty horizontálně vedle sebe. Na společné desce je umístěna v přední části mechanika standu, v zadní poté elektronika. Mechanickou část tvoří motory, propojené dvěma spojkami a poddajnou hřídelí. Vzhledem k malé setrvačnosti motorů jsou na hřídeli připevněny dva setrvačníky tak, aby bylo možné dosáhnout požadované nízké vlastní frekvence soustavy. Bez setrvačnicků by poddajnost systému zajišťovala pouze poddajná hřídel, jejíž průměr by musel být velmi malý a délka nepřiměřeně velká. Po instalaci setrvačnicků se rozměry hřídele navýší, pro oporu setrvačných hmot a odlehčení zátěže drobným ložiskům motorů je nutné instalovat ložiskové domky. Klade se tím větší důraz na přesnou montáž a vzniká nebezpečí, že se soustava bude chovat více jako čtyřmotorový systém. Projevit se mohou také pasivní odpory, a to z ložiskových domků vlivem odvalování ložisek, případně vlivem nesouososti všech komponent. Schématické rozložení komponent je naznačeno níže (Obrázek 5-3):



Obrázek 5-3 – Varianta 2

Elektronická část standu obsahuje dvojici měničů, napájecí zdroj a PLC WAGO PFC200. Výhodou je zde snadné propojení obou komor, protože na rozdíl od sériové struktury není mezi oběma částmi nutná tuhá deska. Kompatibilitu programů napsaných

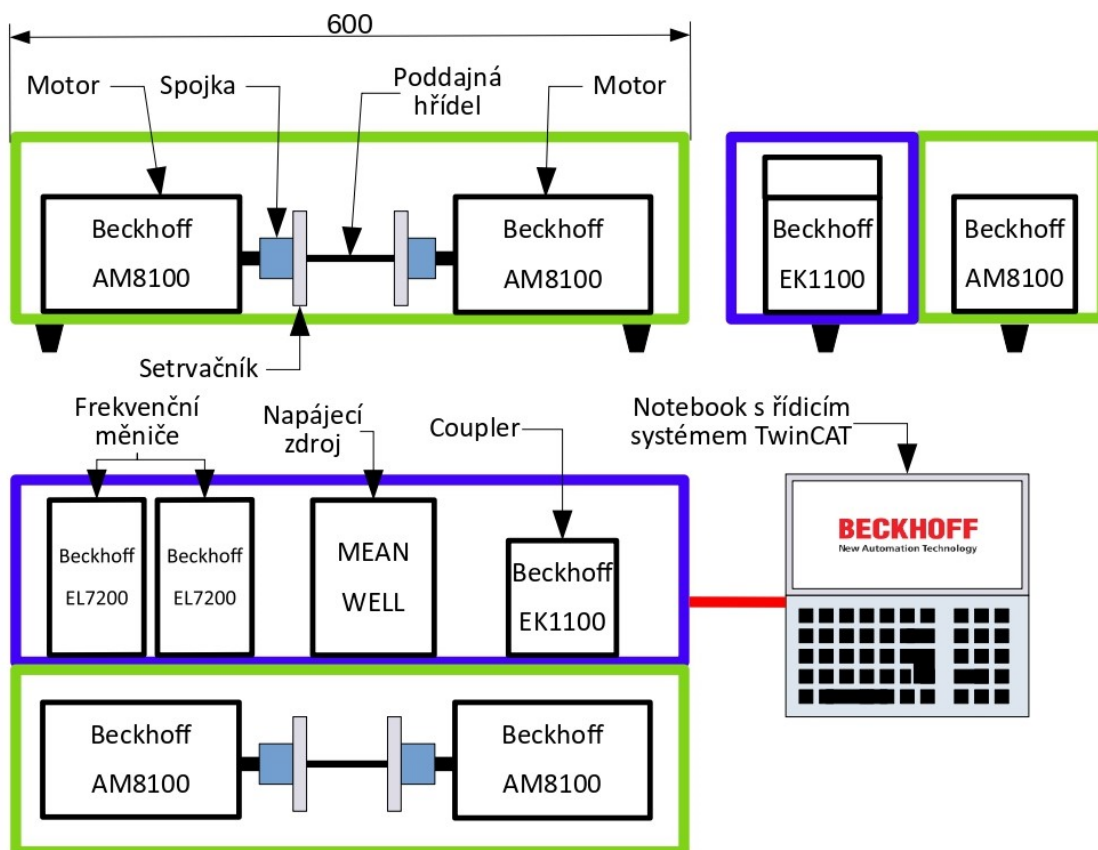
v REXYGEN Studio s ostatními vývojovými prostředími zajišťuje možnost použití rozšířeného programovacího jazyka Function Block Diagram. Alternativou je programování ve vlastním grafickém prostředí, jenž lze propojit se Simulinkem od MathWorks pomocí knihovny RexLib. Tím lze opět získat přístup k pokročilým vývojovým knihovnám a usnadnit tím následný vývoj vlastní regulace.

### 5.3 Varianta 3 – Beckhoff s vlastními pohony

Třetí varianta je kompletně postavena na komponentech Beckhoff. Řídicí systém XAE zde běží na notebooku paralelně s operačním systémem Windows 10 na dedikovaném jádře. Dražší alternativou by bylo použití průmyslového počítače (IPC) od firmy Beckhoff, například C6017. Doporučované takty komunikace s periferiemi I/O se dle výrobce pohybují okolo 0,5 až 1 ms (1–2 kHz). Tento takt by měl být dostatečný pro možnou implementaci rychlostní a polohové zpětné vazby přímo do řídicího systému. Výhodou běhu na počítači běžné architektury je jeho snadná dostupnost v době nedostatku čipů a dlouhých dodacích lhůt hardwaru. Může zde být paralelně spuštěno vývojové prostředí XAE a notebook může též sloužit jako uživatelské rozhraní HMI. Stand v tomto případě není možné provozovat bez připojeného notebooku/PC.

Komunikace s periferiemi probíhá přes EtherCAT sběrnici, notebook je připojený ethernetovým kabelem ke coupleru EK1100. K němu jsou dle potřeby připojeny další vstupní a výstupní karty. Pohony zde byly rovněž vybrány od společnosti Beckhoff, konkrétně motory z řady AM8100 a měniče EL7200 a tvoří kompaktní celek, v praxi označovaný jako „kartičková serva“. Stejnosměrné bezkartáčové motory (BLDC) disponují výkonem 0,25 kW a krouticím momentem 1,34 Nm při napětí 48 V. Kompaktní měniče EL7200 se zapojují přímo do coupleru EK1100 stejně jako např. karty pro digitální vstupy. S motorem jsou propojeny jedním kabelem s využitím one cable technology (OCT). Proudová regulační smyčka běží v taktu 32 kHz, rychlostní poté v taktu 16 kHz. Součástí motoru je brzda. Komunikace a nastavení pohonů probíhá skrz nástroj Drive Manager 2, ve kterém je možné definovat a měnit všechny konstanty týkající se pohonu. Je zde také možné nastavit, zda se bude rychlostní a polohová smyčka uzavírat v měniči nebo v řídicím systému, v tomto případě v notebooku.

Rozmístění komponent je u této varianty rovněž horizontálně vedle sebe, mechanická struktura je namontovaná v přední části standu, výkonová elektronika se nachází v části zadní. Motory jsou propojené dvojicí spojek a poddajnou hřídelí. Pro zvýšení setrvačnosti rotorů je ke každé spojce připevněn setrvačník (viz Obrázek 5-4):



Obrázek 5-4 – Varianta 3

Vlastní frekvence soustavy lze měnit jak průměrem a délkou poddajné hřídele, tak šířkou a průměrem setrvačníků. Instalace ložiskových domků by v tomto případě pravděpodobně nemusela být nutná vzhledem k vysoké únosnosti ložisek motorů Beckhoff v radiálním směru. V elektronické části jsou umístěny dva měniče, napájecí zdroj a coupler. Stand musí být vždy připojený k notebooku, na kterém běží řídicí systém.

Propojení Beckhoffu s vývojovým prostředím Matlab zajišťují knihovny TC3 Target for Simulink, TC3 Interface for MATLAB/Simulink a Simulink PLC Coder. Umožňují snadnou implementaci modelů z Matlabu nebo Simulinku a tím využití pokročilých funkcí dostupných v tomto softwaru. V případě potřeby je možné program z Matlabu převést do strukturovaného textu, ve kterém se PLC od Beckhoffu běžně programují, a tento kód nadále upravovat a ladit ve vývojovém prostředí XAE na počítači, který není připojený k servopohonům.

## 5.4 Porovnání variant a výběr vhodného řešení

V následující podkapitole byly porovnány všechny tři varianty. Pro výběr vhodného řešení byla popsána jednotlivá kritéria, na základě kterých došlo k ohodnocení standů na stupnici od 1 do 5, větší množství bodů značí lepší řešení. Porovnávací kritéria zohledňují jak mechanickou stavbu standu, tak možnosti programování pohonů a kompatibilitu vývojového prostředí s dalšími nástroji.

Prvním porovnávacím kritériem je konstrukce celého standu. Mělo by se jednat o kompaktní přenosný celek, který bude moci transportovat jedna osoba. Rám standu by měl být jednoduchý, aby bylo snadné v případě potřeby vyrobit větší množství zařízení. V tomto ohledu získává body horizontální koncepce vedle sebe variant 2 a 3, která oproti první vertikální variantě umísťuje všechny komponenty na jednu desku. Provedení by mělo být také stabilnější kvůli níže položenému těžišti a mělo by docházet k nižšímu ovlivnění pohonů stoupajícím teplem z výkonové elektroniky. Elektronická a mechanická část je zde lépe přístupná. Mezi výhody sériové struktury varianty 1 patří možnost pozorovat pohony z více stran, je zde ovšem nutné použít dvě tuhé desky – jednu na umístění elektroniky, druhou pro montáž pohonů. Motory jsou v této variantě propojeny pouze poddajnou hřídelí, což může komplikovat případné naladění soustavy na nízkou vlastní frekvenci. V případě potřeby je možné uvažovat o přidání setrvačných hmot ke spojkám stejně jako bylo učiněno u třetí varianty. U druhé konstrukční varianty s ložiskovými domky hrozí riziko, že se soustava bude chovat kvůli tuhosti spojek více jako čtyřmotový systém, čímž by se zkomplikovalo naladění na požadované vlastní frekvence.

Dalším kritériem pro porovnání jsou samotné řídicí systémy. LabVIEW je v průmyslu používáno nejen pro laboratorní úlohy a měření, ale bývá také nasazován v kombinaci s hardwarem NI do průmyslových řídicích procesů. Provoz s pohony ovšem není tak běžný a společnost National Instruments spoléhá na propojení se systémy třetích stran. REXYGEN je v průmyslu obecně rozšířen velmi málo, v kombinaci s pohony ho ovšem úspěšně použili výzkumníci Západočeské univerzity v Plzni [26]. Výhodou je zde snadná tvorba řídicího programu v grafickém prostředí a podobnost se Simulinkem.

Opačná situace nastává u Beckhoffu, který se specializuje na průmyslová řešení řízení pohonů. Princip komunikace pohonů s řídicím systémem odpovídá běžným standardům používaným v praxi a ladění regulačních smyček probíhá stejně jako u reálných strojů. Pro měření a vyhodnocování naměřených dat jsou dostupné speciální knihovny. Problém může nastat v případě, že je nutné měřit/vyhodnocovat data jiným způsobem, než pro který byly knihovny vytvořeny.

Kompatibilita s dalšími vývojovými prostředími, primárně s Matlabem a Simulinkem, je zajištěna u všech variant. Beckhoff vyniká množstvím knihoven a způsobů propojení s dalšími systémy. Mezi netradiční způsoby patří například možnost generování PLC kódu z Matlabu se syntaxí přímo pro Beckhoff. LabVIEW je ovšem v mnoha případech v měření soběstačný, měřené signály je možné vyhodnotit přímo ve vývojovém prostředí, které je pro takové úkoly připravené. REXYGEN umožňuje propojení pouze se Simulinkem a Siemensem.

V kritériu výpočetní výkonnosti jasně vede první varianta se zařízením cRIO disponující programovatelným hradlovým polem FPGA. Takty tohoto pole jsou bezkonkurenční, jedná se o vhodnou platformu pro vývoj nových řídicích algoritmů. TwinCAT ve třetí variantě spoléhá na výkon procesoru notebooku a spolehlivě umožňuje takty komunikace okolo 2 kHz, což by mělo být pro řízení pohonů rovněž přijatelné. U REXYGENu ve druhé variantě je doporučený takt cyklu 200 Hz, což by už nemuselo být dostatečné pro zpětnovazební smyčky uzavřené mimo měnič.

Důležitým faktorem při výběru vhodné varianty je bezesporu cena celého pracoviště. Vertikální koncepce u první varianty s komponenty na sobě bude pravděpodobně o něco dražší než zbylé konstrukce. Do rozpočtu se ovšem promítne razantněji nutnost nákupu řídicího hardwaru od National Instruments. CompactRIO s vhodnými kartami stojí řádově desetinásobek PLC od WAGO ve druhé variantě. K tomu je nutné přičíst cenu licence LabVIEW a také cenu všech dalších modulů nutných pro provoz, jako jsou NI SoftMotion a LabVIEW Model Interface Toolkit. Pro univerzity nabízí NI akademickou licenci, v rámci které jsou obvykle moduly zahrnuty. V tomto ohledu vychází lépe druhá varianta, cena licence pro REXYGEN je zanedbatelná a není zde nutné dokupovat žádné další moduly.

Dobře zde vychází i třetí varianta postavená na Beckhoffu. Společnost poskytuje vývojové prostředí, run-time a všechny potřebné moduly zdarma ve formě testovací sedmidenní licence, která se obnovuje vždy s novým zadáním CAPTCHA kódu. Některé funkce mohou být v testovacím režimu omezené, například počet vstupů a výstupů importovaného modelu ze Simulinku. Systém je provozován na běžném notebooku, pro propojení s pohony je nutné dokoupit pouze coupler, který stojí stejně jako PLC WAGO z druhé varianty. Ceny pohonů první a druhé varianty jsou přibližně shodné, druhá varianta má měniče a motory zhruba o polovinu levnější. Daní za to je jejich nižší výkon, který by se mohl negativně projevit při překonávání pasivních odporů ložiskových domků.

Výsledné bodové ohodnocení jednotlivých variant bylo zapsáno do tabulky (viz Tabulka 5-1). Kritéria konstrukce, řídicího systému a kompatibility vyhrála třetí varianta postavená na platformě Beckhoff. První varianta s komponenty od National Instruments a Kollmorgen se ukázala být jako nejvýkonější, druhá varianta se systémem REXYGEN a pohony Nanotec/Maxon se naopak projevila jako nejlevnější alternativa. V konečném součtu bodů jasně zvítězila třetí varianta s komponenty Beckhoff. Jedná se tedy o koncepci, na jejímž základě bude v následující kapitole navrhnout stand pro výuku servopohonů.

*Tabulka 5-1 – Porovnání variant*

Kritérium\Varianta	Var. 1 – LabVIEW	Var. 2 – REXYGEN	Var. 3 – Beckhoff
Konstrukce	2	3	4
Řídicí systém	3	3	5
Kompatibilita	4	3	5
Výkon	5	1	3
Cena	3	5	4
<b>Body celkem:</b>	<b>17</b>	<b>15</b>	<b>21</b>



## 6 Návrh standu

V následující kapitole byl popsán návrh konstrukce kompletního pracoviště. Nejprve byla vypočtena vyhovující kombinace tuhosti hřídele a setrvačností rotorů. Po vybrání vhodné velikosti servomotoru bylo přistoupeno k návrhu konstrukce setrvačníků. Následovaly kontrolní výpočty pohonné části a návrh konstrukce nosné struktury standu, která sestává z hliníkové desky a rámu. Pro motory byla vybrána příslušná výkonová elektronika včetně měničů a napájecího zdroje. V závěru kapitoly je popsáno zajištění bezpečnosti pracoviště a to jak pro obsluhu, tak pro samotné elektronické komponenty.

### 6.1 Výpočet modálních vlastností pohonu

Jak již bylo zmíněno v předešlých částech práce, jedním z požadavků na stand je nízká vlastní frekvence rotační soustavy. Při běžném návrhu stroje se konstruktér obvykle snaží o dosažení co nejvyšší hodnoty vlastní frekvence. Toho může dosáhnout buď zvýšením výsledné tuhosti navrhovaného pohonu, nebo snížením jeho setrvačných hmot, jak již vyplývá ze známého vztahu:

$$\Omega = \sqrt{\frac{k_t}{J}} \quad (6-1)$$

V praxi do návrhu pohonu negativně vstupuje řada faktorů, jako je například poddajnost kuličkového šroubu, řemenového převodu, spojek, setrvačné hmoty a poddajnost planetových převodovek, poddajnost hřídelí a mnoho dalších činitelů. Výsledkem je obvykle poměrně nízká vlastní frekvence soustavy, která se promítne do možností naladění kaskádní regulace. Velikost propustného pásma pohonu s poddajnou mechanikou je nižší než standartní hodnota. Stand pro výuku servopohonů by měl tuto skutečnost reflektovat, byla proto požadována co nejnižší vlastní frekvence pohonu.

Okrajové podmínky při výpočtu tuhosti a setrvačností tvoří primárně požadavek na maximální šířku standu 600 mm a z toho vyplývající volná délka poddajné hřídele 100 mm. Požadovaná vlastní frekvence byla stanovena na 70 Hz. Dosažení takto nízké vlastní frekvence by mělo být zajištěno použitím poddajné hřídele o malém průměru a případně přidáním setrvačných hmot k rotorům servomotorů.

Vhodnými a dostupnými kandidáty na pohony jsou motory Beckhoff z řady AM8100, konkrétně modely AM8121, AM8131 a AM8141. Jedná se o stejnosměrné bezkartáčové šestipólové motory s napájecím napětím 48 V, které se liší především nominálním krouticím momentem  $M_k$  a setrvačností rotorů  $I_{motor}$ . Při výpočtu modálních vlastností byl nejprve stanoven pro každý motor minimální průměr poddajné hřídele  $d_{min}$  na základě znalosti dovoleného napětí  $\tau_d$  pro střídavý krut u oceli ČSN 11 600 a bezpečnosti  $k_{bezKR} = 3$ :

$$d_{min} = \frac{16 \cdot M_k}{\sqrt[3]{\tau_d \cdot \pi} \cdot \sqrt{k_{bezKR}}} \quad (6-2)$$

Po výpočtu byl minimální průměr hřídele zaokrouhlen vždy na nejbližší vyšší normalizovanou hodnotu. Délka poddajné hřídele  $l$  byla volena s ohledem na požadavek šířky standu do 600 mm tak, aby se vešel na běžný stůl i v kolmém směru. Do rovnice byl doplněn modul pružnosti ve smyku  $G$  a byla dopočtena torzní tuhost hřídele:

$$k_t = \frac{G \cdot \pi \cdot d_{min}^4}{32 \cdot l} \quad (6-3)$$

Následoval výpočet setrvačnosti hmot tak, aby bylo dosaženo požadované vlastní frekvence. Soustava je aproximována dvoumotovým uvolněným systémem. První vlastní frekvence tedy činí 0 Hz, druhou vlastní frekvenci je možné určit pomocí vztahu vycházejícího z antirezonančních frekvencí:

$$\Omega = \sqrt{(\omega_1^*)^2 + (\omega_2^*)^2} = \sqrt{\frac{k_t}{J_1} + \frac{k_t}{J_2}} \quad (6-4)$$

Při uvažování stejné setrvačnosti pohonů lze vztah zjednodušit a vyjádřit z něj celkovou požadovanou setrvačnost jedné hmoty  $J_{celk}$ :

$$J_{celk} = \frac{2 \cdot k_t}{\Omega^2} \quad (6-5)$$

V dalším kroku byla odečtením setrvačnosti rotoru vypočítána požadovaná dodatečná setrvačnost, čímž bylo potvrzeno, že je nutné instalovat setrvačníky:

$$J_{setrvačník} = J_{celk} - J_{motor} \quad (6-6)$$

Rozměry setrvačníku jsou omezeny jak maximálním poloměrem  $r_{vnější}$  s ohledem na velikost standu, tak minimálním poloměrem  $r_{vnitřní}$  s uvažováním potřebného průměru v setrvačnicku pro upevnění. Zbývá dopočítat šířku setrvačnicku  $l_{setrvačnick}$ :

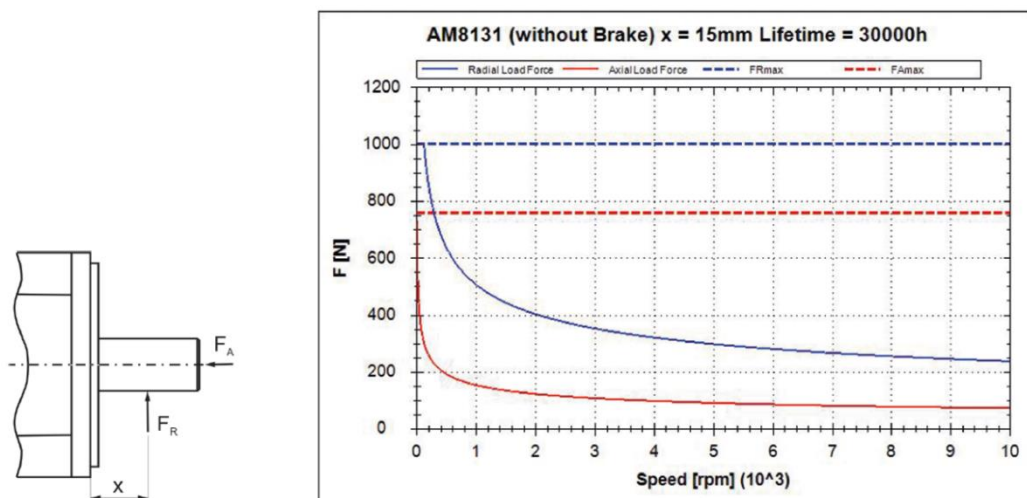
$$l_{setrvačnick} = \frac{2 \cdot J_{setrvačnick}}{\rho_{ocel} \cdot \pi \cdot (r_{vnější}^2 - r_{vnitřní}^2) \cdot (r_{vnější} + r_{vnitřní})} \quad (6-7)$$

Pro výpočet statického zatížení motorů byla vypočtena radiální síla od hmotnosti setrvačnicku působící na hřídel pohonu:

$$F_{rad} = m_{setrvačnick} \cdot g = \rho_{ocel} \cdot l_{setrvačnick} \cdot \pi \cdot (r_{vnější}^2 - r_{vnitřní}^2) \cdot g \quad (6-8)$$

Tato síla byla porovnána s katalogovými hodnotami zatížení motorů. Beckhoff poskytuje software Beckhoff Load Force Calculator, ve kterém je možné stanovit dovolené zatížení při volitelné poloze radiální síly  $x$  a libovolné požadované životnosti. Je zde také možnost zvolit motor s mechanickou brzdou. Jako příklad je zde uvedeno dovolené radiální zatížení motoru AM8131 (Obrázek 6-1). [32]

Výsledky výpočtů byly zapsány do Tabulky 6-1. Vstupními parametry byla volná délka poddajné hřídele (100 mm), požadovaná vlastní frekvence (70 Hz), setrvačnost motoru a vnitřní a vnější průměr setrvačnicku (46 mm, respektive 100 mm). Výstupními parametry byl minimální průměr hřídele a délka setrvačnicku, jenž odpovídá požadavku na jeho setrvačnost. Skripty pro výpočet parametrů jsou součástí přílohy „Konstrukcni\_parametry\_AM81x1.m“. Koeficient bezpečnosti  $k_{nezpečnost}$  byl vypočten jako podíl únosnosti motoru a radiální síly od setrvačnicku při uvažovaném těžišti setrvačnicku vzdáleném 15 mm od příruby, otáčkách 1000 ot./min a požadované životnosti 20 000 hod.



Obrázek 6-1 – Dovolené radiální zatížení motoru AM8131 [33]

Pro motor AM8141 nejsou data o únosnosti k dispozici, ale vzhledem k použití rozměrnějších ložisek než u motoru AM8131 se předpokládá únosnost stejná nebo vyšší. Motor AM8121 sice dosahoval vysoké životnosti vlivem nízkého zatížení od setrvačnicku, nevýhodou byl ale nízký nominální krouticí moment 0,5 Nm. Existuje zde riziko, že chování takovéto soustavy bude příliš ovlivněno pasivními odpory, proto byl motor z volby vyloučen. Opačná situace nastala u motoru AM8141. Vlivem relativně vysokého krouticího momentu bylo nutné zvětšit průměr hřídele. Důsledkem toho byl nárůst rozměrů setrvačnicků, což by vedlo ke zvětšení celkové konstrukce standu. Jako vhodná volba se tedy jeví střední výkonnostní řada AM8131 s vypočteným průměrem hřídele 8 mm a šířkou setrvačnicku 51,6 mm. Zvolen byl motor AM8131-1J21-0000.

*Tabulka 6-1 Přehled základních konstrukčních parametrů pro vybrané motory*

Parametr \ Motor	AM8121	AM8131	AM8141
$d_{min}$ [mm]	6 mm	8 mm	9 mm
$J_{motor}$ [kgm <sup>2</sup> ]	$13,4 \cdot 10^{-6}$	$46,2 \cdot 10^{-6}$	$108 \cdot 10^{-6}$
$J_{setrvačnick}$ [kgm <sup>2</sup> ]	0,0012	0,0038	0,0060
$l_{setrvačnick}$ [mm]	16,3 mm	51,6 mm	82,1 mm
$F_{radiální}$ [N]	7 N	23 N	36 N
$k_{bezpečnost}$ [-]	47	29	>18,75
$\Omega$ [Hz]	70	70	70
$\omega^*$ [Hz]	49,5	49,5	49,5

## 6.2 Návrh setrvačnicků a spojky

Z modálních výpočtů rotační soustavy jasně vyplynula nutnost použití přídavných setrvačných hmot k rotorům servomotorů. Tyto hmoty je možné instalovat přímo na hřídel servomotoru, na pružnou spojku nebo na poddajnou hřídel. Z hlediska zatížení motorů je ideální první varianta. V tomto případě by ovšem bylo nutné navrhnout spojku pro poddajnou hřídel jako součást setrvačnicku, a to vzhledem k jejich potřebné šířce. Proto bylo od této varianty ustoupeno. V případě umístění setrvačnicku na poddajnou hřídel hrozil nepřiměřený průhyb hřídele a vyšší zatížení ložisek motorů, protože se těžiště hmoty nachází dále od příruby motoru. Vhodné by tedy bylo připevnění setrvačnicků ke spojkám, propojujícím servomotor a poddajnou hřídel.

### 6.2.1 Návrh spojky

Hřídelových spojek se na trhu nachází velké množství. Proto byla provedena rešerše nabízených produktů a byly vybrány varianty, které by se pro použití v rámci standu nejlépe hodily. Spojky by měly být dostatečně tuhé, aby příliš neovlivňovaly poddajnost soustavy. U spojek s nižší tuhostí by se mohly projevovat nelinearity v průběhu tuhosti a jejich únosnost by nemusela být dostatečná.

První navrhovaná varianta je miniaturní bezvúlová vlnovcová spojka TOOLFLEX MINI. [33] Tvoří ji náboje z hliníku a prostřední vlnovcová část z ušlechtilé oceli, viz Obrázek 6-2:



*Obrázek 6-2 – Vlnovcová spojka*

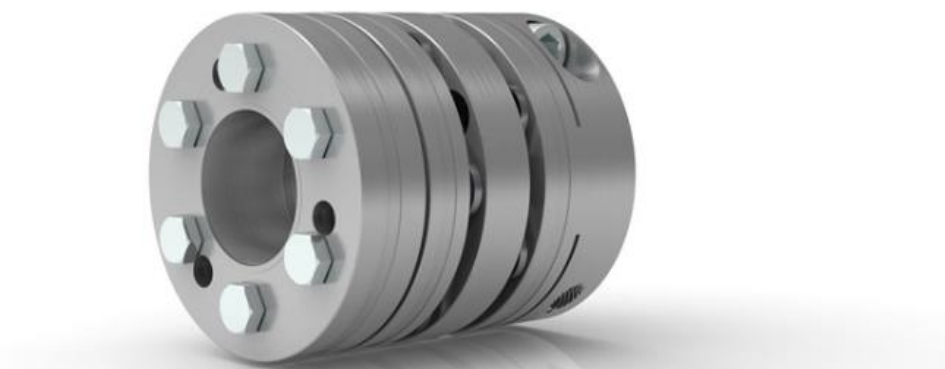
Vlnovec je tuhý v torzním směru, ale velmi poddajný při axiálním a radiálním vyosení hřídelí. Toto je výhodné u běžné aplikace, protože nemusejí být kladeny tak vysoké požadavky na přesnou montáž. Při použití dvojice vlnovcových spojek a poddajné hřídele na standu ale hrozí, že se celý systém při roztočení vlivem setrvačné hmoty prohne a celek začne kmitat nejen torzně, ale i radiálně, což je pro tuto aplikaci nežádoucí. Dostupná a konstrukčně jednoduchá alternativa je spojka korýtkového typu. [34] Svěrná verze je tvořena dutým válcem, který je na jedné straně naříznutý a v této části je stažený čtveřicí šroubů viz Obrázek 6-3:



*Obrázek 6-3 – Svěrná korýtková spojka [35]*

Spojka se vyrábí také v dělené verzi. Tělo spojky je rozdělené na dvě části, které jsou stažené šrouby na obou stranách. Korýtková spojka nabízí vysokou tuhost ve všech směrech, je ale hůře dostupná v menších rozměrech a výrobní tolerance bývají nízké, proto se běžně v servopohonech nevyskytuje.

Dalším nabízeným typem je bezvúlová lamelová servospojka, někdy označovaná jako kardanová vkládací spojka. [34] Tvoří ji dva ocelové náboje, mezi které je umístěna pružná lamela. Hlavní výhodou této verze jsou její kompaktní rozměry, mezi další benefity patří vysoká tuhost v krutu, únosnost a pro tuto aplikaci i vyšší ohybová tuhost oproti vlnovcové spojce. Prodává se v provedení s jednou nebo se dvěma lamelami. Níže je zobrazen tzv. dvojitý kardanový typ (Obrázek 6-4):



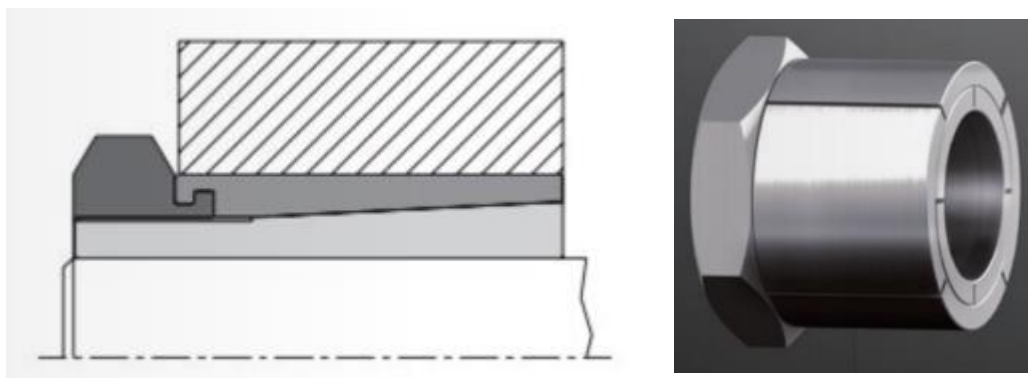
*Obrázek 6-4 – Lamelová spojka kardanová [36]*

Lamelová spojka se jeví z navrhovaných variant jako nejlepší volba vzhledem k jejím kompaktním rozměrům a vysoké tuhosti. Protože bude použita v páru, může být zvolena jednolamelová varianta. Celá soustava se pak bude chovat jako homokinetický kloub, stejně jako kdyby byla použita jedna dvojitá kardanová spojka mezi motory bez poddajné hřídele. V ustáleném stavu tedy bude tedy zajištěna konstantní úhlová rychlost obou rotorů. Vybrána byla spojka RADEX NC 16/EK/2.5-8/2.6-14 od firmy KTR. Náboje spojky jsou duralové, pružná lamela je vyrobena z oceli. Setrvačnost spojky činí  $50 \cdot 10^{-6} \text{ kgm}^2$ , torzní tuhost odpovídá  $20 \cdot 10^3 \text{ Nm/rad}$ .

### 6.2.2 Návrh setrvačnicku

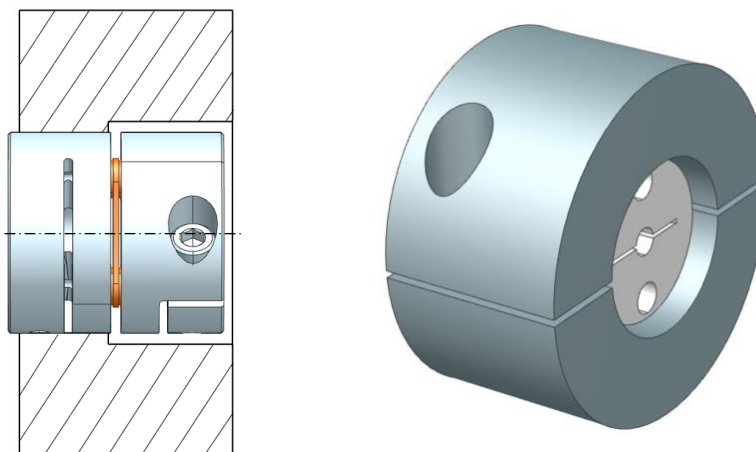
Jak již bylo uvedeno v úvodu této kapitoly, setrvačnický by bylo žádoucí připevnit přímo ke spojkám, v ideálním případě k náboji hřídele motoru. Tím bude zajištěno že by po zanedbání poddajnosti motorové hřídele tvořil rotor a setrvačnický jednu setrvačnou hmotu. Setrvačnický může být ke spojce připevněn svěrně, stavěcími šrouby, případně šroubovým spojením z čela spojky.

První varianta je konstrukčně velmi jednoduchá. Setrvačnický tvoří ocelový disk, který je připevněn ke spojce pomocí samostředícího svěrného pouzdra, které se používá k upevnění nábojů na hladkou hřídel (viz Obrázek 6-5):



*Obrázek 6-5 – Uložení setrvačnicku na spojku svěrným pouzdem*

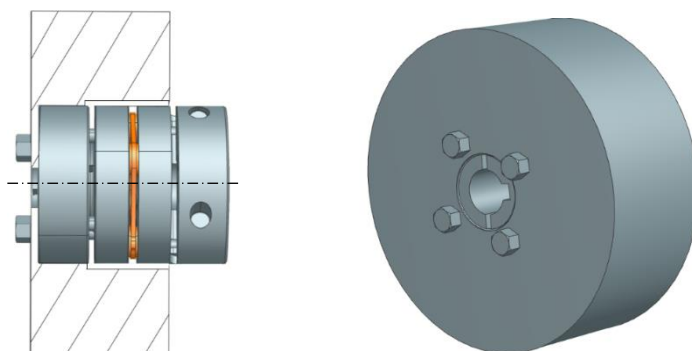
Překážku zde klade především délka pouzdra s maticí, která roztahuje svěrné klíny. Výrazně menší rozměry má setrvačnický, který sestává ze dvou částí stažených šrouby, podobně jako u dělené korýtkové spojky. Takovýto setrvačnický může dosedat pouze k nábojové části spojky a ve zbytku prostoru může být odlehčený (viz Obrázek 6-6):



*Obrázek 6-6 – Uložení setrvačnicku na spojku svěrně šrouby*

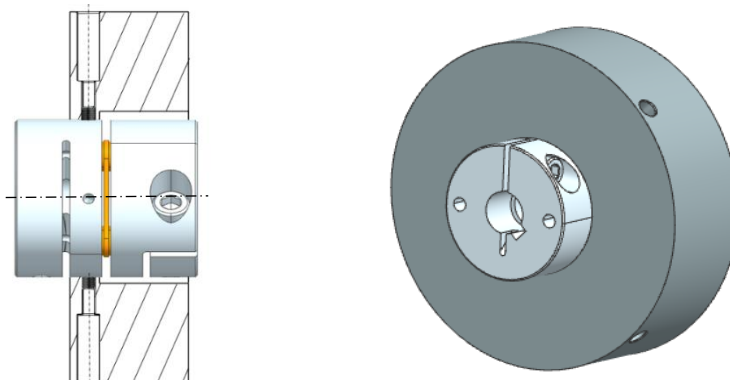


Třetí varianta zahrnuje setrvačnick přitažený šrouby k čelu spojky. Setrvačnick je vystředěn na její nábojovou část u motoru (viz Obrázek 6-8). V oblasti planžet a hřídelového náboje je setrvačnick opět odlehčen tak, aby nedocházelo ke kontaktu. Výhodou je zde setrvačnick tvořený jedním kusem, který vyplňuje maximální dostupný prostor okolo pružné spojky. Problém nastává při montáži. Nejprve musí být dotažena spojka a až poté připevněn setrvačnick. Pohodlnému dotažení šroubů setrvačnicku brání příruba motoru. Proto by musely být v setrvačnicku vytvořeny servisní otvory pro dotahování spojky, což celou konstrukci komplikuje a stává se tak nevýhodnou.



*Obrázek 6-7 – Uložení setrvačnicku na spojku čelními šrouby*

V poslední variantě řešení je disk nasunutý na nábojovou část spojky s přechodným uložením (viz Obrázek 6-9). Zajištěn je čtveřicí stavěcích šroubů po obvodu. Část setrvačnicku nad lamelou a hřídelovým nábojem je opět odlehčena. Setrvačnick je umístěn tak, aby bylo možné spojku nasadit na hřídel bez nutnosti jeho demontáže. Vyplnění dostupného prostoru je v této variantě rovněž efektivní. Hrozí zde ale stříh stavěcích šroubů při nouzovém zastavení motoru pomocí integrované brzdy. Po zvážení všech uvedených aspektů a na základě předešlé zkušenosti ze stavby standů byla nakonstruována varianta svěrného uložení setrvačnicku na spojku pomocí dvojice šroubů M10. Výkres setrvačnicku DP1835-01-01 je součástí přílohy.



*Obrázek 6-9 – Uložení setrvačnicku se zajištěním stavěcími*



## 6.3 Kontrolní výpočty

Po dokončení mechanického návrhu pohonu bylo možné přistoupit ke kontrolním výpočtům. Pro tyto účely byl v Matlabu vytvořen skript s názvem „Kontrolni\_vypocty\_pro\_motor\_AM8131.m“. Nejprve byly aktualizovány modální vlastnosti soustavy, následoval kontrolní výpočet minimálního průřezu šroubu, otlacení mezi hřídelí a objímkou, kontrola tlaku v dosedací ploše pod hlavou šroubu a kontrola tlaku v závitech. Pro šrouby byl stanoven potřebný utahovací moment.

### 6.3.1 Ověření modálních vlastností

Použitím pružných spojek došlo ke změně celkové poddajnosti a setrvačnosti soustavy, a tedy i ke změně druhé vlastní frekvence. Protože se jedná o tuhosti v sérii, byla k převrácené hodnotě tuhosti poddajné hřídele z rovnice 6-3 přičtena převrácená hodnota tuhosti spojek z katalogu:

$$k_{\text{celková}} = \frac{1}{\frac{1}{k_{\text{spojka}}} + \frac{1}{k_t} + \frac{1}{k_{\text{spojka}}}} = 358,25 \text{ Nm/rad} \quad (6-9)$$

Dále byla ve výpočtu požadované setrvačnosti setrvačnicku z rovnice 6 uvažována setrvačnost spojky:

$$J_{\text{setrvačnick}} = J_{\text{celk}} - J_{\text{motor}} - J_{\text{spojka}} = 0,0036 \text{ kgm}^2 \quad (6-10)$$

Výsledná požadovaná setrvačnost jednoho setrvačnicku činí 0,0036 kgm<sup>2</sup>. Z této hodnoty vyplývají další parametry setrvačnicku – vnitřní průměr 46 mm, vnější průměr 100 mm a šířka 54 mm. Výrobní výkres setrvačnicku je součástí přílohy. Rozměry poddajné hřídele zůstaly nezměněné – průměr 8 mm a volná délka 100 mm.

### 6.3.2 Kontrola svěrného spojení

Velikost působícího momentu  $M_{k\text{svěrné}}$  na setrvačnick byla uvažována jako trojnásobek nominálního krouťícího momentu motoru (1,3 Nm). Bezpečnost svěrného spojení  $k_{\text{svěrné}}$  byla zvolena 3, délka svěrného spojení setrvačnicku s objímkou spojky  $l_{\text{svěrné}}$  činí 8 mm na průměru spojky  $d_{\text{spojky}}$  46 mm. Součinitel tření  $f$  mezi spojkou a setrvačnickem odpovídá koeficientu pro ocel 0,12. Celková svěrná síla byla vypočtena dle vztahu:

$$Q = \frac{\pi \cdot k_{\text{svěrné}} \cdot M_{k\text{svěrné}}}{4 \cdot f \cdot d_{\text{spojka}}} \quad (6-11)$$

Při použití dvou šroubů je svěrná síla  $Q_{02}$  poloviční. Dovolené tahové napětí  $\sigma_{kt}$  pro šrouby pevnostní třídy 8.8 odpovídá 630 MPa. Minimální průřez šroubu lze určit vztahem:

$$S_3 = \frac{Q_{02}}{0,7 \cdot \sigma_{kt}} \quad (6-12)$$

Výsledný malý průměr šroubu  $d_3$  odpovídá hodnotě 1,3 mm a je výrazně menší než u použitých šroubů M10 s malým průměrem 8,16 mm:

$$d_3 = \sqrt{\frac{4 \cdot S_3}{\pi}} = 1,3 \text{ mm} \ll 8,16 \text{ mm} \quad (6-13)$$

### 6.3.3 Kontrola na otláčení

Nejprve bylo zkontrolováno svěrné spojení mezi setrvačником a spojkou na plošné otláčení  $p_0$ . Napětí je výrazně menší než povolená hodnota pro dural:

$$p_0 = \frac{4 \cdot Q}{d_{spojka} \cdot \pi \cdot l_{svěrné}} = 4 \text{ MPa} \ll 80 \text{ MPa} \quad (6-14)$$

Následovala kontrola tlaku  $p_{hs}$  v dosedací ploše pod hlavou šroubu. Ve výpočtu je počítáno s velkým průměrem závitu M10  $d_1$  a s průměrem hlavy šroubu  $d_{hlavy}$ . Tlak v dosedací ploše bezpečně vyhověl vůči dovolené hodnotě napětí pro ocel:

$$p_{hs} = \frac{Q_{02}}{\pi \cdot \left(\frac{d_{hlavy}}{2}\right)^2 - \left(\frac{d_1}{2}\right)^2} = 4 \text{ MPa} \ll 300 \text{ MPa} \quad (6-15)$$

Posledním kontrolovaným parametrem byl tlak v závitech šroubů  $p_z$ . Pro jeho výpočet je nutné znát střední průměr závitu M10  $d_2$ , stoupání závitu  $p_{závit}$  a hloubku závitu  $h_{závit}$ . Dále je potřeba vypočítat nosnou délku závitu  $H_1$ :

$$H_1 = \frac{d_{hlavy} - d_1}{2} = 3,8 \text{ mm} \quad (6-16)$$

Tlak v závitech s přehledem vyhověl dovolené hodnotě napětí:

$$p_z = \frac{Q_{02}}{\pi \cdot d_2 \cdot H_1 \cdot \frac{h_{závit}}{p_{závit}}} = 2 \text{ MPa} \ll 200 \text{ MPa} \quad (6-17)$$

Zbývalo stanovit potřebný utahovací moment šroubů. Moment pro dotažení šroubů na setrvačnicku vychází ze vztahu:

$$M_{ks} = \frac{Q_{02} \cdot f \cdot \frac{d_{hlavy} + d_1}{2}}{2} \quad (6-18)$$

K utahovacímu momentu šroubů je nutné přičíst třecí moment šroubu. Pro jeho získání byl nejprve vypočítán úhel stoupání  $\gamma$ , úhel boku profilu  $\beta$  a třecí úhel  $\varphi$  s uvažováním součinitele tření v závitu  $f_z$  rovném 0,15:

$$\gamma = tg^{-1} \left( \frac{p_{závit}}{\pi \cdot d_2} \right) \quad (6-19)$$

$$\beta = tg^{-1}(tg\beta \cdot \cos\gamma) \quad (6-20)$$

$$\varphi = tg^{-1} \left( \frac{f_z}{\cos\beta_n} \right) \quad (6-21)$$

Třecí moment šroubu  $M_{tm}$  se určí na základě vztahu:

$$M_{tm} = Q_{04} \cdot \frac{d_2}{2} \cdot tg(\gamma + \varphi') \quad (6-22)$$

Výsledný utahovací moment šroubu  $M_{kk}$  s respektováním třecího momentu odpovídá 3,2 Nm:

$$M_{kk} = M_{tm} + M_{ks} = 3,2 \text{ Nm} \ll 48,4 \text{ Nm} \quad (6-23)$$

Tato hodnota je výrazně menší než předepsaný utahovací moment pro šrouby M10 pevnosti 8.8 ( $M_{kk} = 48,4 \text{ Nm}$ ). Proto bude setrvačnick dotážen dle doporučené tabulkové hodnoty.

## 6.4 Konstrukce nosné struktury standu

Stand má být využíván primárně ve výuce, čemuž by měla být uzpůsobena i jeho konstrukce. Pracoviště by mělo být možné transportovat jednou osobou a mělo by se vejít na běžný pracovní stůl jak na šířku, tak i v kolmém směru. Obvyklá hloubka stolu činí 600 mm, jedná se tedy o maximální přípustnou šířku standu. Pracoviště by mělo být kompletně zakrytováno, aby nedošlo k úrazu případně poškození mechanické a elektronické části. Servomotory produkují velké množství tepla. Pokud je například simulován řezný proces, jeden motor působí momentem jako řezná síla a druhý pracuje proti této síle jako vřeten. Vzniklý rozdíl krouticích momentů způsobuje výrazné ohřátí motorů. Proto je nutné zajistit efektivní ventilaci standu a kontrolu vnitřní teploty.

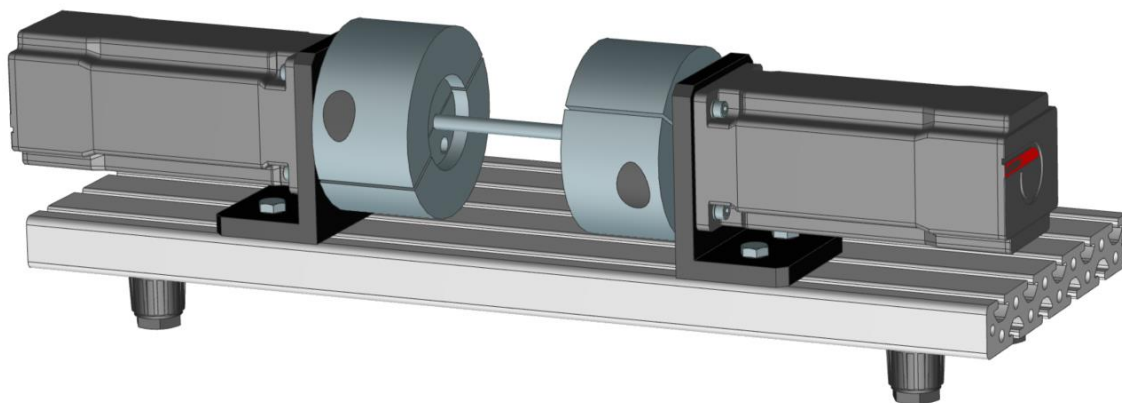
### 6.4.1 Rám

Základ standu je tvořen hliníkovou deskou o rozměrech 28 x 200 x 540 mm s drážkou šířky 8 mm o hmotnosti 5,5 kg. Materiál desky EN AW-6063 je dostatečně lehký v porovnání s běžnou ocelí a zároveň poskytuje potřebnou tuhost. Upevnění motoru bylo vyřešeno držáky MISUMI, jejichž rozměry lze konfigurovat přímo v internetovém konfigurátoru. Motor je upevněn s přechodným uložením  $\varnothing 60$  H7/j6 a zajištěn čtveřicí šroubů ISO 4762 M5x16 s podložkou 5.3 ISO 7089. Předepsaný utahovací moment šroubů činí 5,5 Nm. Konzole motoru je možné objednat pod objednávacím kódem, který rovněž obsahuje její rozměry:

LAFWD-SSB-T10-A100-B60-L80-N8-MA5-D60-F53-G53-H60-P20-S35-V40-X40

Konzole je k hliníkové desce připevněna dvěma šrouby se šestihrannou hlavou DIN 933 M8x20 a maticemi M8 do profilů BH 30-8 s kolmou pružinou a vedením. Deska stojí na přístrojových nožkách s nastavitelnou výškou. Nožky jsou zhotoveny z termoplastu vyztuženého skelnými vlákny, vnitřní mechanismus je ocelový. Výrobce je německá firma Norelem a prodávají se pod kódovým označením 27840-300808. Připevněny byly opět pomocí matic M8 do profilů BH 30-8.

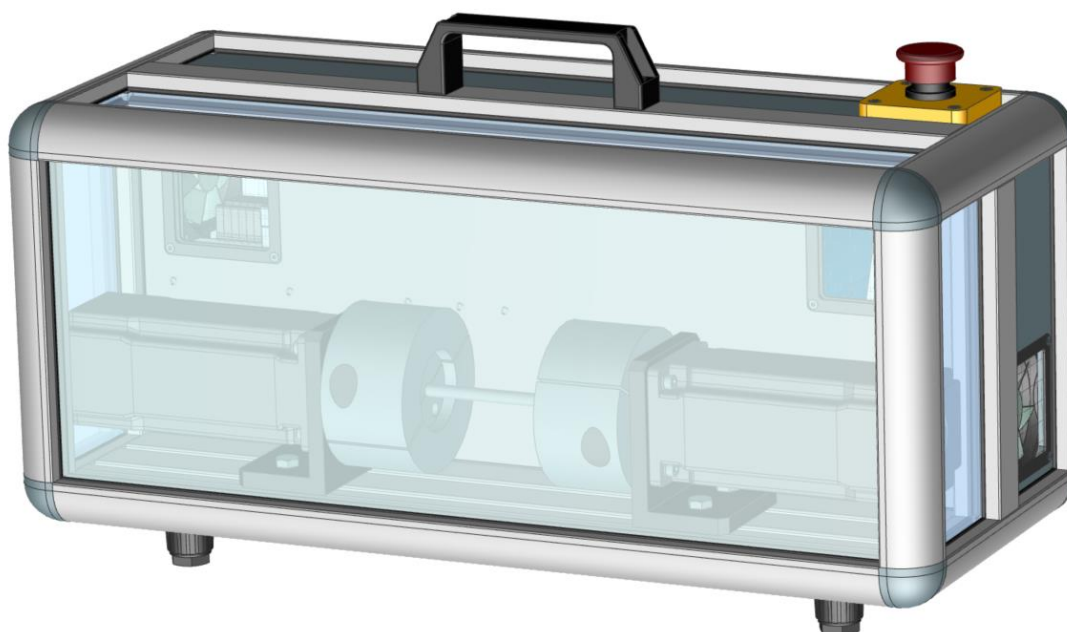
Servomotory jsou propojeny spojkami KTR RADEX NC 16/EK/2.5-8/2.6-14 a poddajnou hřídelí o průměru 8 mm a délce 140 mm z oceli ČSN EN 14 060 (Obrázek 6-10). Volná délka hřídele činí 100 mm. Uložení spojky na hřídeli motoru je tvarové s využitím těsného pera, axiálně je zajištěna stahovacím nábojem spojky. Uložení poddajné hřídele mezi motory je řešeno rovněž svěrně stahovacími náboji spojek.



*Obrázek 6-10 – Pohon na hliníkové desce*

Okolo nosné desky byl navržen rám z profilů ze slitiny hliníku EN AW 6060 T5. Hlavní část rámu je tvořena půlkulatými profily BH 30-8 s rozměry 30 x 30 mm a délkou 8x 200 mm a 4x 540 mm. Spojeny jsou třísými rohovými spojkami pro profily BH 30-8 s kulatou krytkou ze šedého polyamidu. Pro propojení rámu s hliníkovou deskou byly v jejím čele na každé straně vytvořeny vždy dva závity M6. Do těchto závitů byly našroubovány čtyři šrouby s kladívkovou hlavou M6x20 do drážky BH 30-8. Rám byl na šrouby ze strany v půlce jeho kompletování nastrčen. Pro oddělení prostoru s elektronikou od mechanické části bylo nutné vytvořit uprostřed standu příčku z hliníkových profilů BH 30-8 se třemi drážkami o rozměrech 30 x 30 mm. Délka prostředního profilu činí 540 mm, dvojice postranních profilů měří 200 mm. Upevnění bylo opět zajištěno šrouby ISO 7380 M8x25. Pro snadný transport bylo k prostřednímu profilu připevněno madlo z polyamidu o rozměrech 146 x 44 mm. Při montáži byly použity šrouby s válcovou hlavou DIN 7984 M8x16 a matice M8 do drážky profilu.

Prostřední přepážka byla vytvořena z lesklého černého extrudovaného plexiskla o tloušťce 5 mm. Plexisklo bylo řezáno laserem. Ze stejného materiálu byly vytvořeny rovněž boční stěny oblasti s elektronikou. Mechanická část standu je krytována plným bezbarvým polykarbonátem o tloušťce 4 mm. Ze stejného materiálu byla vytvořena i zadní stěna standu. Polykarbonát byl nařezán frézou na požadované rozměry. Pro těsné uložení krytování do hliníkových profilů byl použit zasklívací profil do drážky o rozměrech 30 x 30 mm. 3D model standu ve formátu .stp je součástí přílohy.

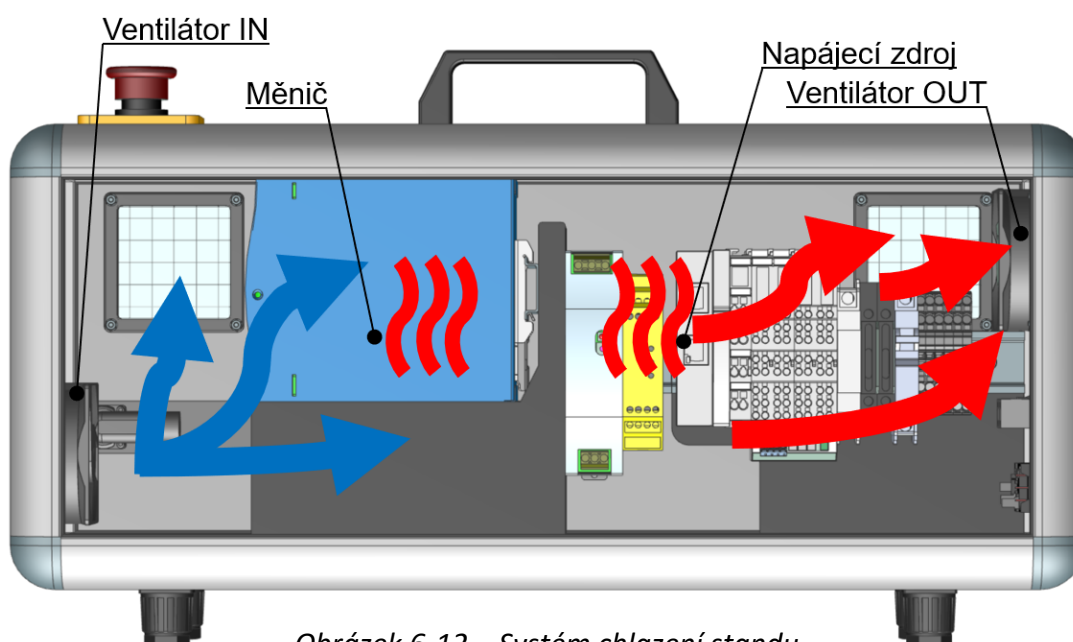


*Obrázek 6-11 – Rám standu s krytovaním a pohonem*

## 6.4.2 Chlazení

Jak již bylo uvedeno v úvodu kapitoly, servomotory generují velké množství tepla. Podobná situace nastává i u výkonové elektroniky. Teplo vzniká v napájecích zdrojích usměrněním střídavého napětí na stejnosměrné, stejně jako ve frekvenčních měničích spínáním a rozpínáním tranzistorů. Proto byly na bočních stranách vytvořeny otvory, osazené axiálními ventilátory Sunon MF80252V2-1000U-A99 napájené 24 V. O krytí se starají hliníkové prachové filtry PRIMECOOLER PC-DFA80B. Tyto filtry kryjí také otvory propojující mechanickou a elektronickou část (viz Obrázek 6-12).

Vzduch je nasát do vnitřní oblasti standu levým dolním ventilátorem. Předpokládá se, že teplý vzduch v mechanické části bude stoupat směrem nahoru propojovacími otvory a bude vytlačován nasávaných studeným vzduchem směrem na druhou stranu standu. Zde je v horní části namontovaný druhý axiální ventilátor, který vyfukuje teplý vzduch z vnitřní oblasti směrem ven, čímž dochází k ochlazení pracoviště. Zapínání a vypínání ventilátorů má na starosti termostat ELKO TER-3F. Jedná se o jednoduchý termostat pro hlídání a regulaci teploty v rozsahu od 0 do +60°C s vestavěným senzorem teploty. Napájen je 24 V a je možné ho upevnit na DIN lištu. Termostat má pevně nastavenou hysterezi 1°C. Při vnitřní teplotě standu do 35°C zůstávají ventilátory vypnuté. Po překročení této hodnoty dojde k jejich zapnutí do té doby, dokud neklesne vnitřní teplota pod 34°C.



Obrázek 6-12 – Systém chlazení standu

## 6.5 Výkonová elektronika

Výkonová elektronika je v případě standu pro výuku servopohonů tvořena především frekvenčními měniči a napájecími zdroji. Měniče byly vybrány s ohledem na použité servomotory. Následoval výběr dostatečně výkonných napájecích zdrojů a coupleru pro propojení s řídicím systémem v notebooku.

### 6.5.1 Měniče EL7221

Motory Beckhoff AM8131 vyžadují pro maximální výkon napájení 48 V a proud 10 A. Tyto požadavky splňují jednobáňové měniče EL7221-9014 od stejnojmenné společnosti. Propojeny jsou s motory pomocí one cable technology (OCT), kdy je v jednom kabelu přítomno napájení, odměřování, napájení brzdy a datový kabel s informací o teplotě motoru. Karta měniče umožňuje připojení dvojice digitálních snímačů koncových poloh pohonu a disponuje vstupem pro safe torque off (STO). Propojení bezpečnostních funkcí bude popsáno v následující kapitole. Součástí karty EL7221 je řízení motorové brzdy.

Pulzní šířková modulace v měniči (PWM) spíná tranzistory s taktom 16 kHz. Proudová regulace běží výhradně v EL7221, a to v pevně stanoveném taktu 32 kHz. Motor může být řízen z řídicího notebooku v momentovém režimu CST (cyclic synchronous torque). V tomto případě je rychlostní a polohová smyčka uzavírána v řídicím systému a do měniče je poslán pouze požadavek na krouticí moment, který odpovídá požadavku na proud do proudové regulace. Rozšířený režim CSTCA (cyclic synchronous torque with commutation angle) umožňuje zadání požadovaného krouticího momentu včetně komutačního úhlu.

Motor může operovat rovněž v rychlostním režimu CSV (cyclic synchronous velocity). Rychlostní smyčka je uzavřená v kartě měniče s pevně daným taktom 16 kHz a polohová smyčka v řídicím systému s taktom odpovídajícím komunikaci periferií po sběrnici EtherCAT. Tento takt lze teoreticky nastavit až na 4 kHz, v praktickém použití se využívají hodnoty okolo 1-2 kHz. Vyšší takty komunikace umožňují integrované hodiny uvnitř měniče (distributed clocks), které se synchronizují s hodinami v řídicím systému každých 100 ns. CSV režim je základní režim, ve kterém je motor po instalaci obvykle provozován.

Třetí dostupný režim se nazývá CSP (cyclic synchronous position) a umožňuje polohové řízení motoru uzavřené v kartičkovém měniči. V NC úloze v řídicím systému se vypočítává požadovaná trajektorie, která se v taktu komunikace periférií I/O posílá do měniče. V něm jsou uzavřeny všechny tři smyčky kaskádní regulace.

Karty byly doplněny o doporučené aktivní chlazení ZB8610, které se instaluje na spodní část a vyžaduje napájení 24 V. Bez chlazení není možné využít jejich maximální výkon. Průtok vzduchu zajišťuje ventilátor o maximálních otáčkách 5 500 ot./min a možností provozu ve třech režimech. První je aktivní, pokud je k chlazení připojeno pouze napájení. Využívá interního teplotního čidla a termostatu. Při dosažení teploty 40°C se spustí chlazení na poloviční otáčky a s rostoucí teplotou se zvyšují otáčky až do maximálních při teplotě vyšší než 55°C. Ventilátor zůstává spuštěný do poklesnutí teploty pod 35°C. Druhý režim se aktivuje přivedením 24 V na vstupní pin, ventilátor je v ten moment spuštěn na maximální otáčky. Třetí režim umožňuje regulovat otáčky ventilátoru na základě frekvence spínání vstupu ventilátoru. Aktivní chlazení ZB8610 bylo ve standu zapojeno v prvním režimu.

### 6.5.2 Periferie a napájecí zdroje

Propojení mezi řídicím systémem TwinCAT v notebooku DELL a kartičkovými měniči EL7221-9014 zajišťuje coupler EK1100 se sběrnici EtherCAT, využívající ethernetový kabel. Coupler vyžaduje napájení 24 V a je schopen poskytnout maximální proud 10 A. Beckhoff má v nabídce rovněž novější verzi EK1101, kterou je možné připojovat za běhu řídicího systému (tzv. HotConnect). Tato verze ale vyžaduje propojení s řídicími IPC od Beckhoff a není možné ji načíst běžným notebookem, proto nebyla ve standu použita.

O napájení dvojice měničů se stará síťový zdroj Mean Well TDR-960-48, který poskytuje 48 V stejnosměrného napětí a maximální proudu 20 A. Motory vyžadují napájení 48 V a proud 8 A, při paralelním zapojení je tedy výkon zdroje dostatečný. Napájení ostatních elektronických komponent na 24 V obstarává druhý zdroj Mean Well NDR-75-24 s maximálním proudem 3,2 A.



## 6.6 Bezpečnost pracoviště

Hlavní využití standu se nachází ve výuce. Proto je nutné zajistit dostatečnou ochranu obsluhy, a to i pro případ nevhodně nastavených regulačních parametrů a z toho vyplývajícího neočekávatelného chování. Dále je nutné myslet na ochranu elektronických komponent v případě jejich defektu nebo poruchy elektroinstalace.

### 6.6.1 Ochrana obsluhy

Základním pasivním prvkem ochrany je polykarbonátové krytování celého standu o tloušťce 4 mm. Mezi přednosti tohoto materiálu patří vysoká odolnost vůči nárazům, nízká hořlavost a odolnost proti UV záření. Krytování zabraňuje kontaktu obsluhy s pohyblivými částmi standu a s elektronickou instalací. Zároveň chrání obsluhu v případě selhání setrvačnicků. Soubory pro výrobu krytování ve formátu .dwg jsou součástí přílohy. Aktivní ochranu představuje tlačítko nouzového zastavení YW1B-V4E02R od firmy IDEC umístěné v horní části standu. Zapojeno je dvoukanálově do bezpečnostního relé AD SRE4C od společnosti REER, do kterého vede rovněž STO vstup do kartičkových měničů EL7221. Pokud není na vstupu safe torque off (STO) napětí, měnič bezpečně odpojí jakýkoliv přívod proudu do výkonových tranzistorů a tedy i do motoru. Jedná se o hardwarové odpojení, měnič není možné restartovat v řídicím systému do té doby, dokud není opět přivedeno napětí na STO vstup. V případě nebezpečí obsluha stiskne tlačítko pro nouzové zastavení, dojde k rozpojení dvoukanálového obvodu do bezpečnostního relé, které reaguje rozpojením jednonálového obvodu do STO vstupu karet. Přeruší se přívod proudu do motoru a automaticky se aktivuje brzda motoru, čímž se motor zastaví.

### 6.6.2 Ochrana elektrického vybavení

Elektrická instalace standu je chráněna jističem Noark Ex9BN B10 100006. Jistič s charakteristikou B je dimenzován na 10 A při napětí 230 V. Mezi napájecím zdrojem a ostatní elektronikou jsou umístěny pojistky. Pro okruh s napětím 24 V byla použita skleněná trubičková pojistka 5x20mm F 3,15A/250V s maximálním proudem 3,15 A, pro okruh s napětím 48 V pojistka s maximálním proudem 20 A. Pojistky jsou upnuty skrze svorky na DIN lištu. Zapojení je patrné z blokového schéma, které je součástí přílohy „DP1835-Blokove-schema-elektroinstalace.pdf“.

## 7 Návrh řídicího projektu

Řídicí systém na platformě Beckhoff TwinCAT tvoří jádro celého standu pro výuku servopohonů. Běží na notebooku DELL a stará se o komunikaci mezi motory a obsluhou. Pro základní pohyby motorů byly vytvořeny funkční bloky, které se spouštějí pomocí uživatelského rozhraní HMI. Součástí řídicího systému je projekt Drive Manager 2, v rámci kterého lze nastavit parametry motorů a ladit regulační konstanty. Disponuje rovněž osciloskopem zobrazujícím aktuální pohyby motorů. Motory Beckhoff AM8131 a měniče EL7221-9014 byly fyzicky k dispozici po celou dobu programování řídicího projektu. Hotový projekt byl poté odladěn na postaveném standu se setrvačnicí a poddajnou hřídelí.

### 7.1 Instalace Beckhoff TwinCAT

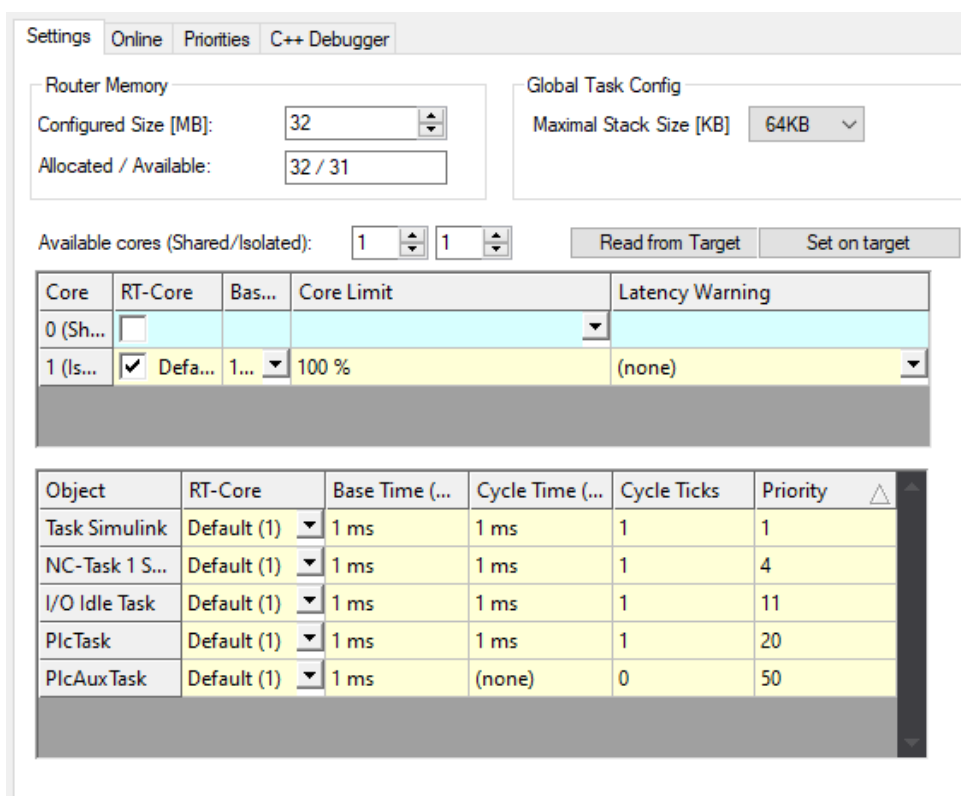
Prvním krokem před instalací řídicího systému byl výběr vhodného kompatibilního počítače. Alternativou by bylo použití IPC od Beckhoffu se systémem Windows, tím by se ovšem celé pracoviště znatelně prodražilo. Vzhledem k mobilitě standu nepřipadaly stolní počítače v úvahu a bylo vybíráno mezi notebooky. Hlavním sledovaným parametrem byla kompatibilní síťová karta. TwinCAT využívá ke komunikaci s periferiemi ethernetový port a vlastní real-time ovladač, který podporuje pouze síťové karty s chipsetem Intel ze seznamu uvedeného na stránkách Beckhoff. Ani přítomnost karty v seznamu ovšem nezaručuje správnou funkčnost, jak se ukázalo u notebooku Lenovo ThinkPad T480 s kartou Intel I219-V. Karta je sice na seznamu kompatibilních zařízení a instalace ovladačů proběhla v pořádku, notebook ale následně nebyl schopen komunikovat s periferiemi, konkrétně s couplerem EK1100.

TwinCAT se podařilo úspěšně zprovoznit až na DELL Latitude E6440 s kartou Intel I217-LM. Notebook využívá starší dvoujádrový procesor i7-4610M se čtyřmi vlákny a 8 GB RAM. Nejprve bylo nainstalováno vývojové prostředí Microsoft Visual Studio 17 s balíčkem Desktop development with C++ pro pozdější implementaci modelu ze Simulinku. Následovala instalace TwinCATu včetně běhového prostředí XAR. Součástí instalace je i načtení vývojového prostředí XAE do Visual Studia. Pokud není Visual Studio k dispozici, je možné instalovat samotné XAE, které je zdarma a je postavené na stejném

jádra jako Visual Studio. Proběhla rovněž instalace modulů pro pozdější použití, konkrétně se jedná o TE2000 HMI Engineering, TE5950 Drive Manager 2, TF3300 TwinCAT 3 Scope Server, TF4100 TwinCAT 3 Controller Toolbox TE1310, TwinCAT 3 Filter Designer a TE1400 Target for Simulink. V posledním kroku byl nainstalován RT ovladač síťové karty prostřednictvím TwinCATu.

## 7.2 Základní nastavení TwinCAT projektu

V Microsoft Visual Studio 2017 byl vytvořen řídicí projekt podle standartní XML šablony TwinCAT XAE Project. V záložce STAND/SYSTEM/Real-Time byl pro vyšší výkon nastaven běh XAR na izolovaném jádře. Takt všech úloh byl nastaven shodně na 1 ms. Nastavení systému je vidět níže (Obrázek 7-1):

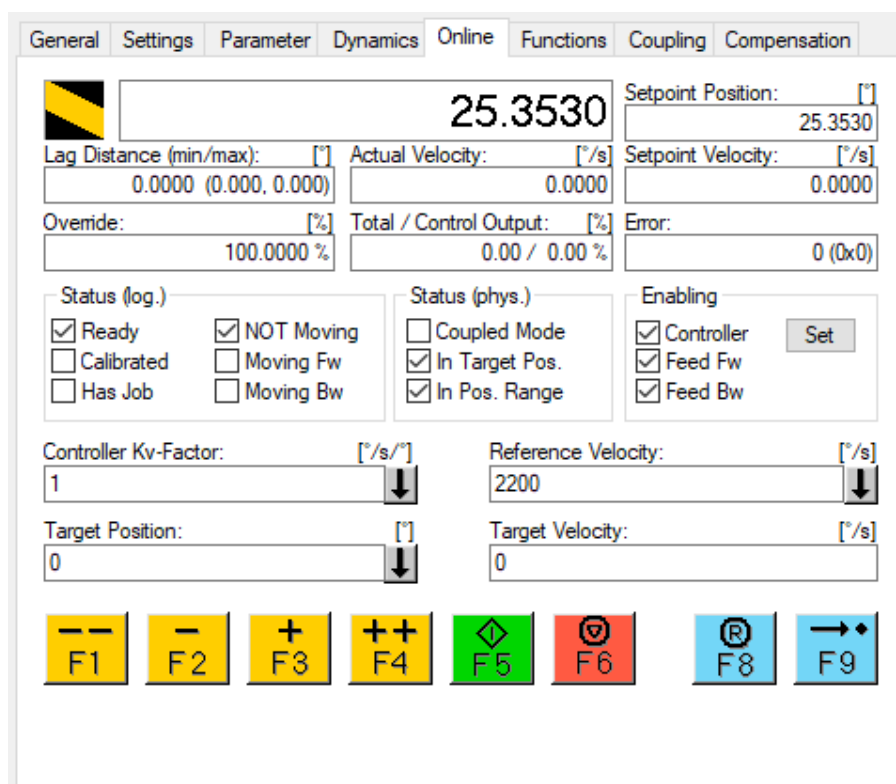


Obrázek 7-1 – Nastavení Real-Time systému

Pro naskenování HW konfigurace, v tomto případě coupleru EK1100 a karet EL7221 s motory AM8131, bylo nutné přejít do Config módu a spustit sken ve stromu STAND/(I/O)/Devices. V NC úloze byly nastaveny základní parametry pohonu – jednotky polohy ve stupních, hodnota maximální rychlosti, zrychlení a jerků. Sledování odchylky požadované a skutečné polohy „Position Lag monitoring“ bylo vypnuto.

Ve Visual Studiu byl vytvořen projekt TwinCAT Drive Manager 2 a načteny NC parametry osy. V záložce „Basic settings“ je možné vidět naskenovaný motor, odměřování polohy a motorovou brzdu. Byla zde nastavena posuvová konstanta na 360°/otáčku (NC feed constant), což odpovídá pohonu bez převodu. V nastavení je možné přidat setrvačnost setrvačnicku. U druhého motoru bylo nutné v záložce „Scaling“ invertovat NC enkodér, protože motory jsou umístěny proti sobě. Volby „Run motor“ a „Tune drive“ umožňují ladění kaskádní regulace a budou dále popsány v kapitole HMI Projekt. Zbývající karta „Diagnostics“ zobrazuje chybové hlášky pohonu a „Advanced“ umožňuje nastavení zobrazování procesních dat. Tato data je pak možné nalinkovat na PLC proměnné a sledovat aktuální hodnoty polohy, momentů a dalších veličin přímo v PLC programu.

Po konfiguraci pohonů lze provádět základní pohyby motoru prostřednictvím NC úlohy. Tato úloha je dostupná v řídicím projektu ve stromu na levé straně pod volbou STAND/MOTION/NC-Task 1 SAF/Axes/Axis 1/Online (Obrázek 7-2):



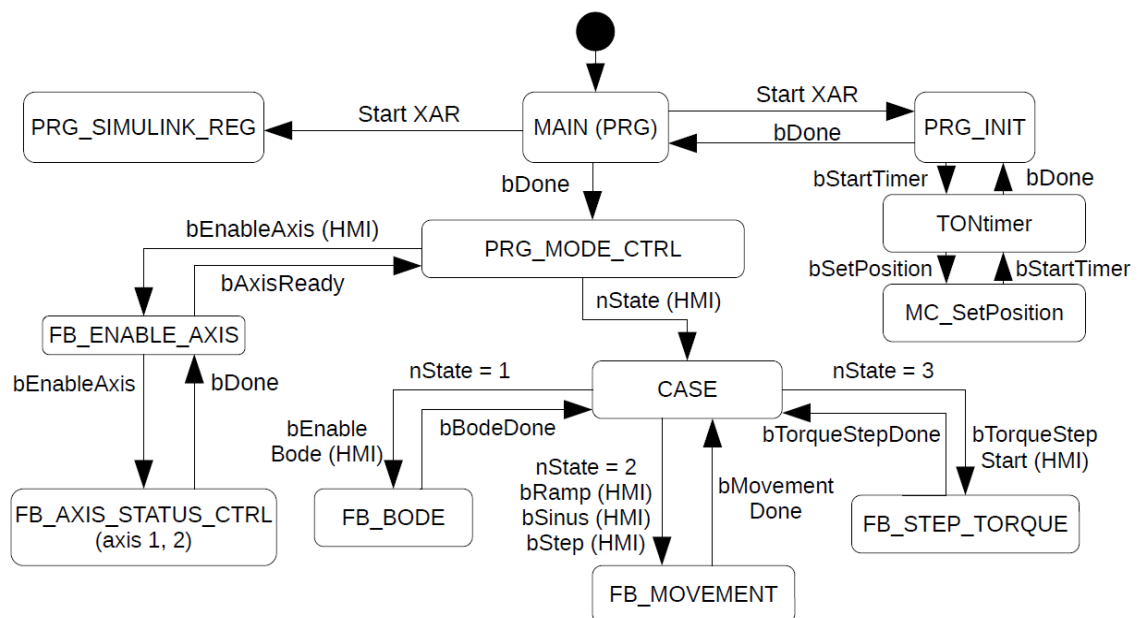
Obrázek 7-2 – Základní pohyb NC osy

Pro osu je nutné povolit ovladač a pohyb v kladném/záporném směru v záložce „Enabling.“ Stand je tvořen dvěma motory propojenými hřídelí, proto je nezbytné spustit oba motory a osy spárovat volbou „Master/Slave Coupling – Linear“ v záložce „Coupling.“ Pokud motor není spuštěný, je aktivována brzda. Po zapnutí režimu Coupling nastaví Slave motor svojí aktuální polohu na polohu Master motoru. Setpoint polohy pro Master motor je automaticky nastavován rovněž jako setpoint pro Slave motor. Oba motory si ponechávají kompletní kaskádní regulaci, jedná se tedy spíše o vazbu Gantry, než o klasický Master/Slave používaný v obráběcích strojích. Master/Slave má polohovou zpětnou vazbu společnou pro oba pohony, které mají k dispozici vlastní rychlostní a proudovou regulační smyčku. Beckhoff v tuto chvíli nemá Master/Slave k dispozici. Motory je možné polohovat klávesami F2 a F3, případně rychloposuvem F1 a F4. Další pohybové funkce se nacházejí pod kartou „Functions,“ jmenovitě například polohování na požadovanou pozici, pohyb po sinusové trajektorii, skok rychlosti nebo konstantní pohyb zvoleným směrem. Volba „Set Actual Position“ umožňuje nastavení aktuální polohy na požadovanou hodnotu.

### 7.3 Struktura řídicího projektu

V předešlé kapitole byly předvedeny základní možnosti ovládání pohonů. Některé úkony se provádějí vždy, když nastane konkrétní událost. Příkladem může být nastavení aktuální polohy pohonů na nulovou hodnotu po spuštění systému, automatický coupling os v lineárním režimu nebo resetování virtuální NC osy před každým enablováním. Tyto procesy byly automatizovány prostřednictvím řídicího projektu. Přidány byly dodatečné funkce pro pohyb, jejichž parametry je možné nastavit v řídicím systému. Akce jsou spouštěny prostřednictvím uživatelského rozhraní HMI. Struktura navrženého řídicího projektu je patrná z vytvořeného stavového diagramu (Obrázek 7-3). Po spuštění systému TwinCAT naběhne program MAIN, který volá program PRG\_INIT pro inicializaci parametrů. Zároveň po celou dobu běhu volá program PRG\_SIMULINK\_REG, který se stará o správný chod regulátoru implementovaného z prostředí Matlab Simulink. Více informací o regulátoru bude uvedeno v samostatné podkapitole.

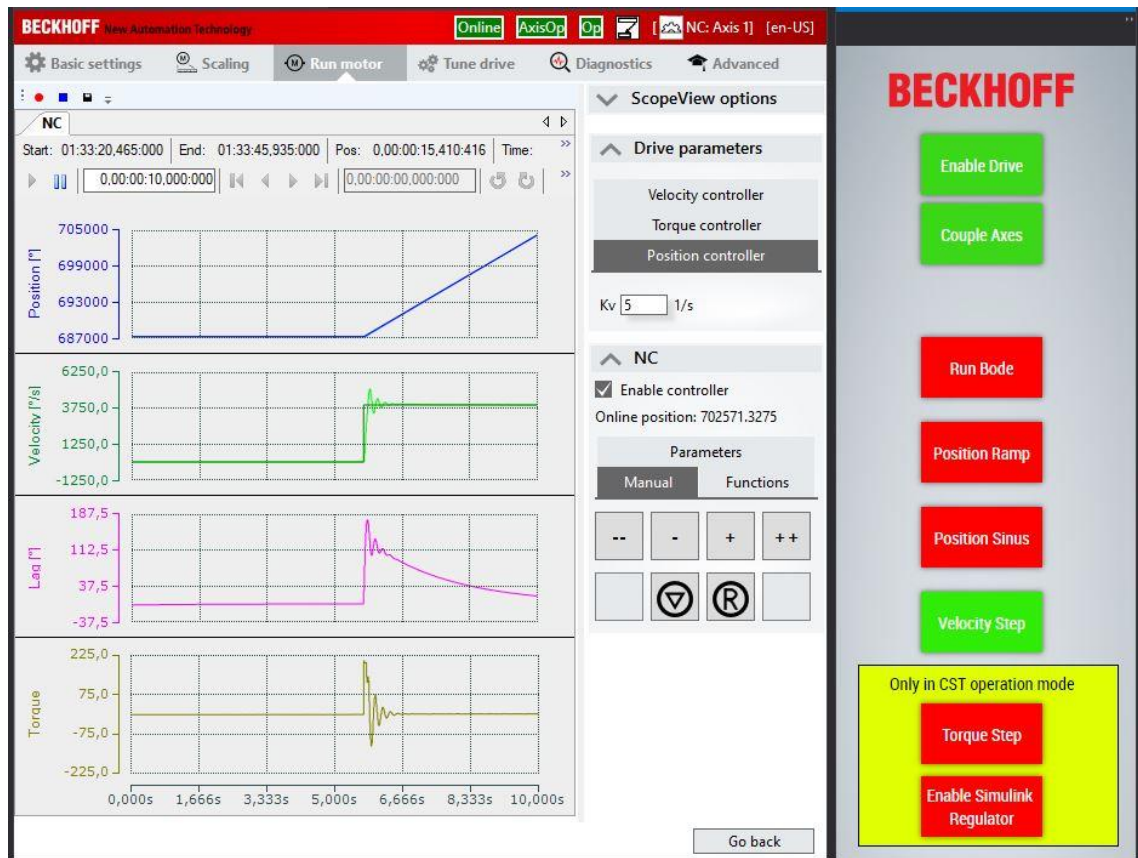
Program PRG\_INIT nejprve spustí časovač TONtimer po dobu jedné sekundy, následně je spuštěn MC\_SetPosition, jenž nastaví polohy pohonů na nulovou hodnotu a opět aktivuje časovač na jednu sekundu. Pokud proběhlo vše v pořádku, výstupní proměnná PRG\_INIT.bDone se přiřadí hodnota TRUE a program MAIN začne volat v každém taktu program PRG\_MODE\_CTRL. V rámci tohoto programu je možné spouštět pohony proměnnou bEnableAxis ovládanou z HMI (viz Obrázek 7-4). Proměnná aktivuje funkční blok FB\_ENABLE\_AXIS, který dá pokyn funkčnímu bloku FB\_AXIS\_STATUS\_CTRL. Pokud resetování a enablování os proběhlo v pořádku, funkční bloky nastaví hodnotu proměnné bAxisReady na TRUE. V případě nastavení proměnné bEnableAxis v HMI na hodnotu FALSE dojde k deaktivování os.



*Obrázek 7-3 – Stavový diagram řídicího projektu*

Součástí programu PRG\_MODE\_CTRL je struktura CASE. Prostřednictvím HMI lze přepínat jednotlivé stavy proměnnou nState. V prvním stavu lze spustit FB\_BODE pro měření dat pro Bode diagram proměnnou bEnableBode. Korektní měření dat je potvrzeno nastavením bBodeDone na TRUE. Ve druhém stavu lze z HMI spustit připravený pohyb servomotoru pomocí FB\_MOVEMENT. K dispozici jsou možnosti pohybu po rampě polohy (bRamp), sinusový průběh polohy (bSinus) nebo skok rychlosti (bStep). Pohyb je aktivován náběžnou hranou příslušné proměnné a běží předem stanovený čas. Doběhnutí signalizuje proměnná bMovementDone.

Poslední třetí stav umožňuje skok momentu s využitím FB\_STEP\_TORQUE. Tento stav je možné využívat pouze v případě přepnutí regulace do momentového režimu CST a manuálním namapování výstupní proměnné nTorqueAx1 z PLC na vstup požadovaného momentu do pohonu Target Torque. Regulátor importovaný ze Simulinku lze rovněž využívat pouze v režimu CST.



Obrázek 7-4 – Drive Manager 2 a vytvořené HMI

## 7.4 Naprogramované funkční bloky

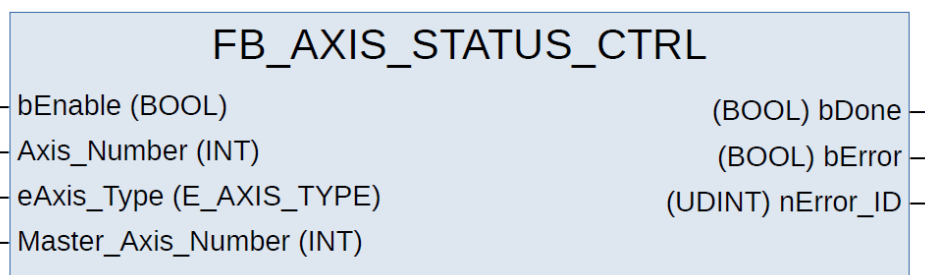
V následujících podkapitolách budou představeny jednotlivé funkční bloky uvedené ve stavovém diagramu řídicího projektu na Obrázku 7-3. Ke každému FB je uvedeno schéma vstupů a výstupů včetně datových typů a stavový diagram. Funkční bloky jsou v PLC projektu uloženy ve složce PLC/STAND\_PLC/FB, globální proměnné se nacházejí v PLC/STAND\_PLC/ GVLs/GVL\_global. Funkční bloky jsou součástí přílohy s řídicím projektem DP1835-TWINCAT-PROJEKT. Jsou zde rovněž uloženy zdrojové kódy ve formátu .pdf.



### 7.4.1 Správa os – FB\_ENABLE\_AXIS a FB\_AXIS\_STATUS\_CTRL

Správu os zajišťuje dvojice funkčních bloků. Aby bylo možné osy ovládat z PLC, bylo nezbytné vytvořit proměnné datového typu `AXIS_REF` a namapovat je k osám v NC úloze. V některých případech je výhodné k osám přistupovat pomocí proměnné integer (INT), proto bylo vytvořeno v `GVL_global` pole os `Axis_Array` typu `AXIS_REF` a proměnné `nAxis_ID_1` a `nAxis_ID_2` typu INT. Komunikace s první osou tedy probíhá zavoláním `Axis_Array[nAxis_ID_1]`, obdobně probíhá komunikace s druhou osou. Pro couplování os byl vytvořen nový datový typ formou enumerace `E_AXIS_TYPE`, která je uložena v `PLC/STAND_PLC/DUTs`. Datový typ určuje, zdali se jedná o osu Master (`eSERVO_MASTER`) nebo o osu Slave (`eSERVO_SLAVE`). Datové typy byly přiřazeny k `GVL_global` proměnným `TypeAx_1` a `TypeAx_2`. Jak již bylo uvedeno, Master/Slave je zde specifické označení výrobce Beckhoff. Zapojení fakticky odpovídá režimu Gantry.

Každá osa má vlastní funkční blok `FB_AXIS_STATUS_CTRL`. Vstupy a výstupy funkčního bloku jsou naznačeny níže (Obrázek 7-5):

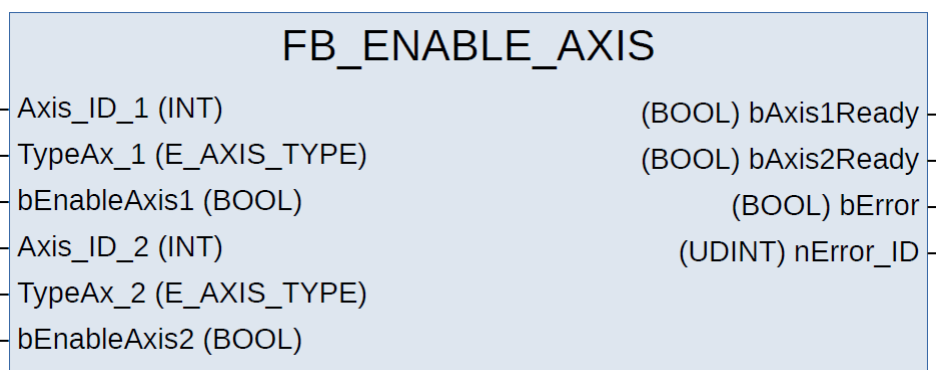


*Obrázek 7-5 – Vstupy a výstupy funkčního bloku FB\_AXIS\_STATUS\_CTRL*

Vstupní proměnná `bEnable` typu `BOOL` odblokuje a aktivuje osu na náběžnou hranu a zablokuje osu na sestupnou hranu. `Axis_Number` udává číslo osy a `eAxis_Type` určuje, zda se jedná o osu Master (může a nemusí být couplována) nebo Slave (po spuštění vždy couplována k ose `Master_Axis_Number`). Výstupní proměnná `bDone` informuje o korektní aktivaci osy. V případě chyby je výstup `bError` nastaven na `TRUE` a `nError_ID` informuje o kódu chyby.



Funkční blok FB\_ENABLE\_AXIS komunikuje s FB\_AXIS\_STATUS\_CTRL a zajišťuje aktivaci/deaktivaci os na základě příkazů z FB\_MOVEMENT. Vstupy tvoří ID první a druhé osy Axis\_ID\_1 a 2, typy osy TypeAx\_1 a 2 a příkazy pro enablování bEnableAxis1 a 2. Výstupy informují o úspěšném enablování os proměnnou bAxis1Ready a bAxis2Ready, chybu zobrazují bError a nError\_ID (Obrázek 7-6).

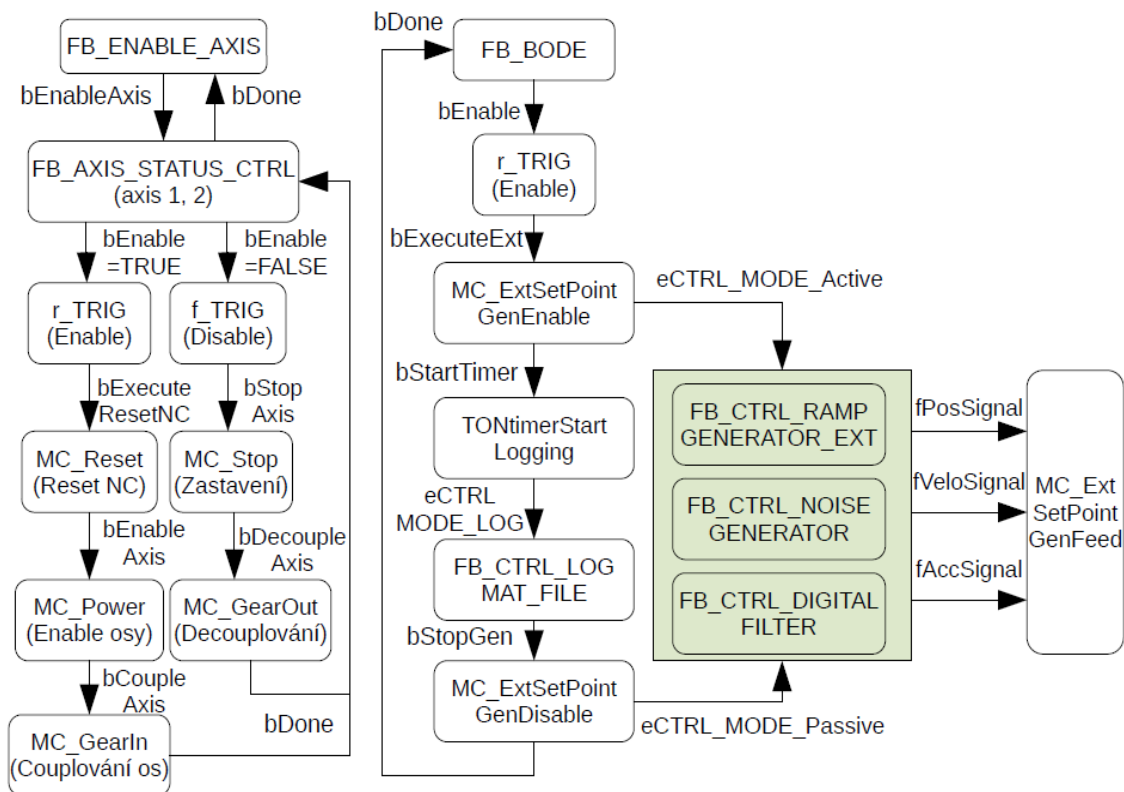


*Obrázek 7-6 – Vstupy a výstupy funkčního bloku FB\_ENABLE\_AXIS*

Komunikace funkčních bloků je znázorněna ve stavovém diagramu (Obrázek 7-8 vlevo). Náběžnou hranou proměnné bEnableAxis je spuštěna aktivace osy v FB\_AXIS\_STATUS\_CTRL. Nejprve dojde k resetování NC osy pomocí MC\_Reset z knihovny Beckhoff, následuje enablování osy blokem MC\_Power. Pokud se jedná o Slave osu, provede se coupling s Master osou pomocí MC\_GearIn. Korektní enablování osy je potvrzeno proměnnou bDone. V případě sestupné hrany bEnable dojde k zastavení a zablokování osy blokem MC\_Stop. MC\_GearOut následně decoupluje osy.

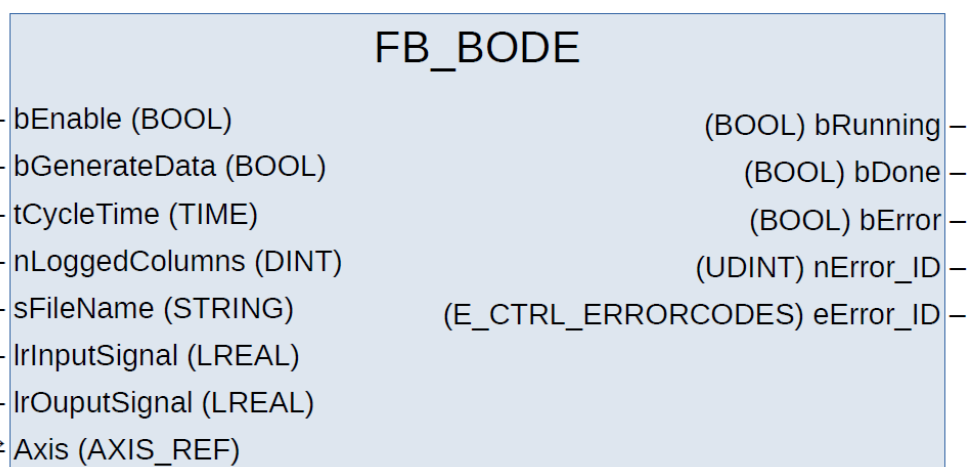
#### 7.4.2 Měření Bode plot – FB\_BODE

Bodeho diagram (Bode plot) umožňuje efektivní diagnostiku a ladění pohonů. Zobrazuje amplitudovou a fázovou charakteristiku v závislosti na budící frekvenci. Pro kaskádní regulaci je důležitá především amplitudová charakteristika přenosu žádané a skutečné veličiny v logaritmickém měřítku. Hranice, při které klesne útlum regulátoru pod -3 dB se nazývá propustné pásmo a jedná se o jeden z ukazatelů kvality naladění regulace. Čím je vyšší propustné pásmo, tím lépe je regulace naladěná. Pro kaskádní regulaci je nejvýznamnější přenos skutečné ku žádané rychlosti, do kterého se promítnou rovněž vlastní frekvence mechanického systému.



Obrázek 7-8 – Stavový diagram FB\_ENABLE, FB\_AXIS\_STATUS\_CTRL a FB\_BODE (vpravo)

Beckhoff nabízí nástroj TE1320 TwinCAT 3 Bode Plot Base, který je ovšem dostupný pouze pro vybrané pohony řady AX5000. Proto byl v rámci této práce naprogramován funkční blok FB\_BODE pro měření amplitudové a fázové charakteristiky Bodeho diagramu (Obrázek 7-7). Spuštěním bloku náběžnou hranou bEnable se pohon Axis (AXIS\_REF) rozjede na offset rychlosti, na který je superponován bílý šum. Offset rychlosti je použit pro zabránění reverzace osy, při které by došlo k negativnímu ovlivnění měření působením pasivních odporů.



Obrázek 7-7 – Vstupy a výstupy funkčního bloku FB\_BODE

Signál polohy, rychlosti a zrychlení je možné generovat (`bGenerateData = TRUE`) nebo použít připravená data v případě taktu úlohy 1 ms (`bGenerateData = FALSE`). Pokud je takt nižší než 1 ms, je nutné data generovat a takt nastavit v proměnné `tCycleTime`. Dle Nyquist–Shannonova vzorkovacího teorému je maximální měřitelná frekvence v Bode diagramu polovina frekvence vzorkovací, což pro takt 1 ms znamená maximální frekvenci 500 Hz. Vstup `lInputSignal` odpovídá žádané veličině, `lOutputSignal` veličině skutečné. Po ustálení přechodového jevu se začnou data nahrávat až do hodnoty `nLoggedColumns`. Data se po naměření uloží do lokace `sFileName` ve formátu `.mat`. Vyhodnocení naměřených dat probíhá v programu Matlab. Výstupní proměnná `bRunning` informuje o běhu měření, `bDone` o dokončení měření a `bError` o chybě. Chybové kódy je možné načíst prostřednictvím `nError_ID` a `eError_ID`.

Obrázek 7-8 zobrazuje v pravé části stavový diagram bloku `FB_BODE` pro případ generování signálu polohy (`fPosSignal`), rychlosti (`fVeloSignal`) a zrychlení (`fAccSignal`). Pokud jsou použity vektory předgenerovaných dat, jsou místo těchto hodnot dosazeny hodnoty `aPos[i]`, `aVelo[i]` a `aAcc[i]`. Proměnná “*i*” se v každém cyklu iteruje. Náběžná hrana vstupní proměnné `bEnable` spustí `MC_ExtSetPointGenEnable`, který umožňuje externí vstup polohy, rychlosti a zrychlení do osy a nahrazuje jimi hodnoty z interního NC interpolátoru. V dalším kroku je spuštěn `FB_CTRL_RAMP_GENERATOR_EXT` generující rampu polohy a `FB_CTRL_NOISE_GENERATOR` generující bílý šum. Tyto signály jsou sečteny a numericky derivovány blokem `FB_CTRL_DIGITAL_FILTER`, který umožňuje implementaci diskrétní přenosové funkce. Skript „`Diskretni_derivace_Beckhoff.m`“ pro výpočet přenosové funkce je součástí přílohy. Přenos sestává z filtru dolní propustí (low-pass filter) prvního řádu vynásobeného derivací *s*:

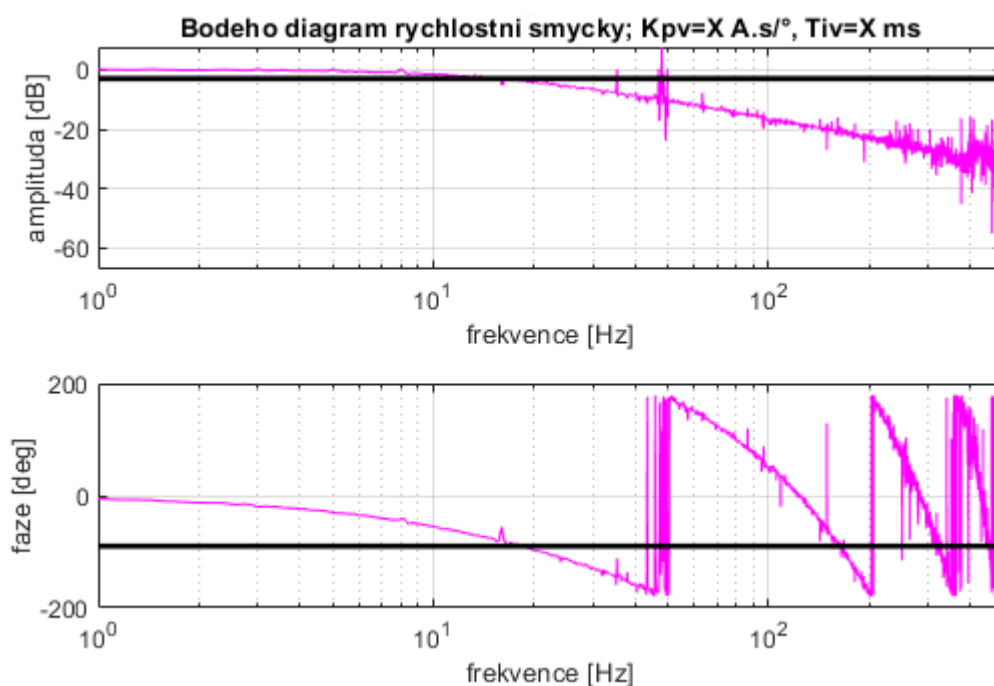
$$G(s) = \frac{s}{\frac{1}{2 \cdot \pi i \cdot f_{lp}} \cdot s + 1} \quad (7-1)$$

Frekvence horní propusti  $f_{hr}$  byla nastavena na 1 000 Hz. Laplaceův obraz přenosu byl transformován do diskrétní oblasti a upraven do DSP formátu:

$$G(z^{-1}) = \frac{1517 - 1517 \cdot z^{-1}}{1 + 0,5171 \cdot z^{-1}} \quad (7-2)$$

Funkční blok FB\_CTRL\_DIGITAL\_FILTER je použit ještě jednou pro derivaci rychlosti na zrychlení. Hodnoty polohy, rychlosti a zrychlení jsou posílány do osy v každém taktu funkcí MC\_ExtSetPointGenFeed. Spuštění externího generátoru aktivuje časovač TONtimerStartLogging. Po uplynutí definovaného času je spuštěno ukládání dat do .mat souboru blokem FB\_CTRL\_LOG\_MAT\_FILE. Po uložení definovaného počtu řádků nLoggedColumns je ukončeno logování dat a soubor je uložen do lokace sFileName. Zároveň je vypnut externí generátor pomocí MC\_ExtSetPointGenDisable.

Naměřená data je možné vyhodnotit přímo na notebooku DELL přiloženým skriptem „Bode\_zpracovani\_dat.m“, používaným na pracovišti RCMT (poskytl vedoucí práce). Příklad naměřených dat je zobrazen níže (Obrázek 7-9):

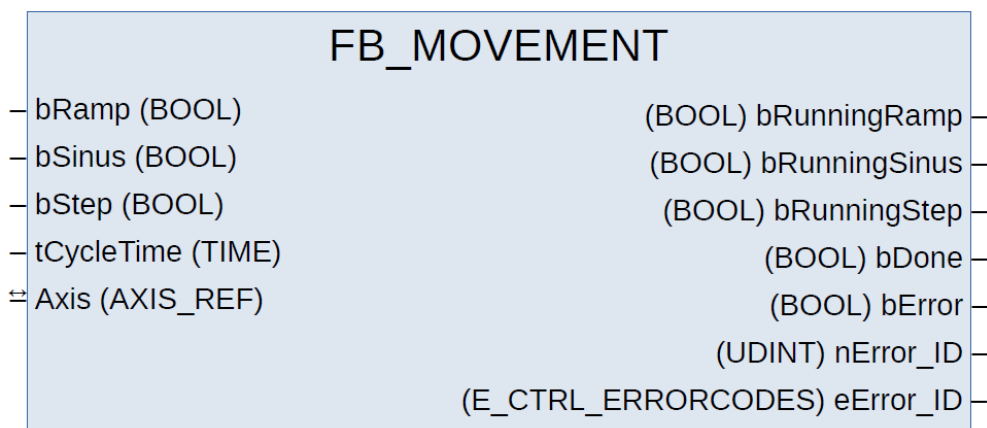


*Obrázek 7-9 – Příklad zpracování dat do Bodeho diagramu*

### 7.4.3 Funkce pro pohyby os – FB\_MOVEMENT

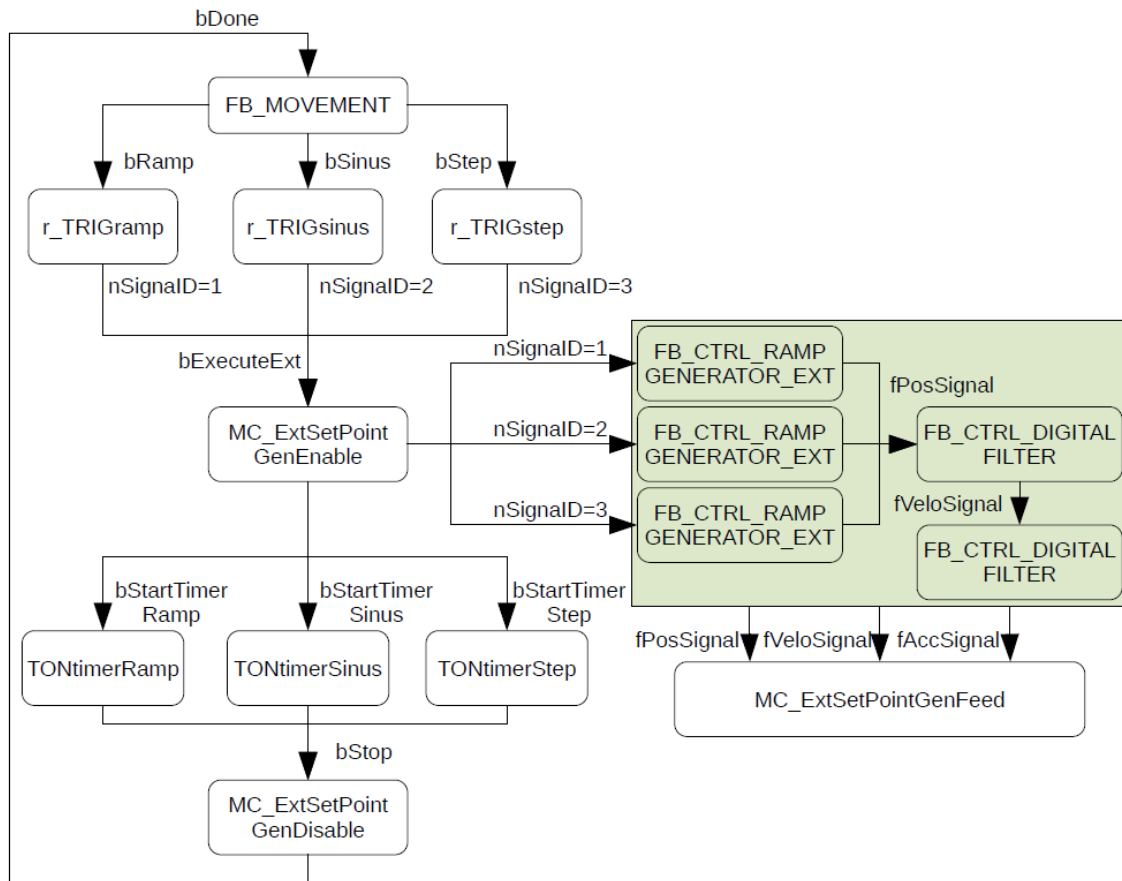
Funkční blok FB\_MOVEMENT umožňuje spojitý pohyb pohonu po předem definované trajektorii (Obrázek 7-10). Pohyb je spuštěn náběžnou hranou jedné z proměnných bRamp, bSinus nebo bStep. Nutné je definovat takt cyklu tCycleTime a přiřadit NC osu Axis. O vykonávání pohybu informují proměnné bRunningRamp, bRunningSinus a bRunningStep. Dokončení pohybu signalizuje bDone. Pokud se vyskytne chyba, informuje o ní proměnná bError a příslušný chybový kód nError\_ID nebo eError\_ID.

Parametry pohybů lze změnit vnitřními proměnnými fPosRampSlope (sklon rampy ve °/s), fPosRampTarget (relativní cílová hodnota ve stupních) a tRampTime (čas běhu). Stejně parametry jsou k dispozici pro skok rychlosti – fStepSlope, fStepTarget a tStepTime. Pohyb po sinusové trajektorii definuje fSinOffset (počáteční offset sinusovky), fSinAmp (amplituda ve stupních), tSinPeriod (perioda sinusovky) a tSinusTime (čas běhu).



Obrázek 7-10 – Vstupy a výstupy funkčního bloku FB\_MOVEMENT

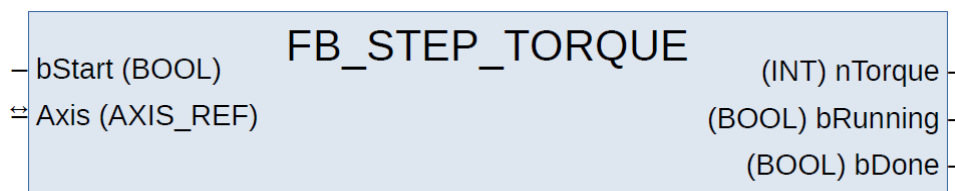
Stavový diagram (Obrázek 7-11) zobrazuje interní komunikaci funkčních bloků, která částečně odpovídá předešlému FB\_BODE. Náběžná hrana vstupní proměnné aktivuje externí vstup signálu do NC osy MC\_ExtSetPointGenEnable a aktivuje časovač TONtimer. Následně je spuštěn generátor polohového signálu na základě zvoleného typu signálu nSignalID. Polohový signál je dvakrát derivován bloky FB\_CTRL\_DIGITAL\_FILTER a výsledné hodnoty jsou posílány do NC osy funkcí MC\_ExtSetPointGenFeed. Po uplynutí zadaného času je generátor deaktivován stejně jako externí vstup signálu do NC osy (MC\_ExtSetPointGenDisable). Proběhnutí pohybu signalizuje proměnná bDone.



Obrázek 7-11 – Stavový diagram FB\_MOVEMENT

#### 7.4.4 Funkce pro skok momentu – FB\_STEP\_TORQUE

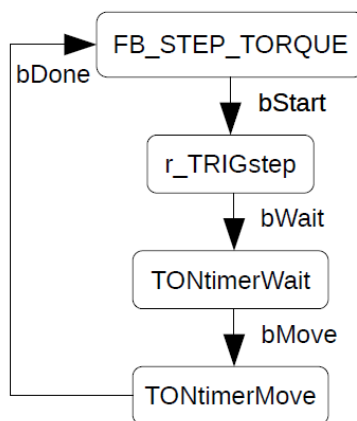
Při ladění proudové regulace Ziegler-Nicholsonovou metodou se sleduje odezva skutečného proudu na požadovaný skok proudu. Proud odpovídá momentu, který je vydělený momentovou konstantou  $K_m$ . Pro účely ladění proudové regulace byl vytvořen funkční blok FB\_STEP\_TORQUE (Obrázek 7-12). Tento funkční blok funguje pouze v režimu continuous torque control (CST), který je možné nastavit v projektu Drive Manager 2 v záložce Tune drive.



Obrázek 7-12 – Vstupy a výstupy funkčního bloku FB\_STEP\_TORQUE

Je rovněž důležité manuálně nastavit výstupní proměnnou nTorque na vstup požadovaného momentu motoru v STAND/(I/O)/Devices/Term 1 (EK1100)/Term 2/(EL7221-9014)/DRV Target torque. Při každé změně režimu regulace se proměnné resetují a je nutné znovu namapovat krouticí moment.

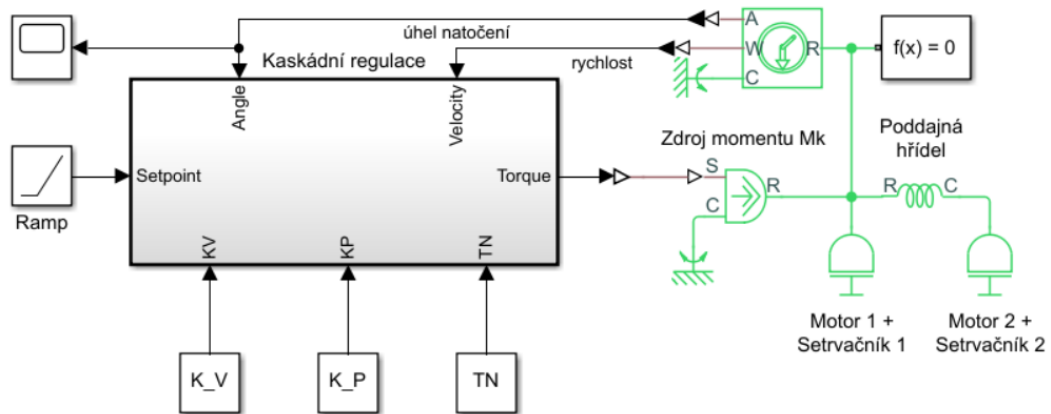
Funkční blok je spuštěn náběžnou hranou proměnné bStart, čímž je aktivován TONtimerWAIT. Po dobu tWait sekund je výstupní nTorque nulový. Následně je aktivován časovač TONtimerMove a moment nTorque je skokově změněn na hodnotu nTorqueSetPoint. Po uplynutí tMove sekund je výstupní moment opět nastaven na nulu a proměnná bDone se změní na TRUE. Jednotlivé kroky jsou znázorněny ve stavovém diagramu (Obrázek 7-13).



Obrázek 7-13 – Stavový diagram FB\_STEP\_TORQUE

## 7.5 Implementace řídicího algoritmu ze Simulinku

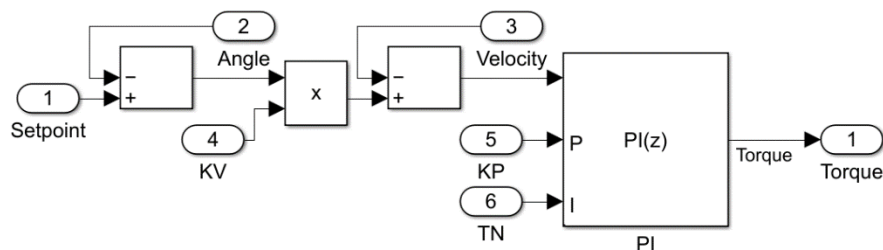
Vývojové prostředí Matlab Simulink nabízí mnoho pokročilých nástrojů pro tvorbu řídicích algoritmů, mezi které patří například Control System Toolbox. Tyto modely je možné exportovat do TwinCATu několika způsoby, které byly uvedeny v rešerši. V praktické části bude představen Simulink PLC Coder a TE1400 TwinCAT 3 Target for Simulink. V Simulinku a Simscape byl nejprve vytvořen model standu pro výuku servopohonů s uvažováním pouze jedné kaskádní regulace (Obrázek 7-14). Druhý servomotor byl nahrazen setrvačnou hmotou. Poddajná hřídel byla zjednodušena na torzní pružinu. Kaskádní regulace je součástí stejnojmenného subsystému.



Obrázek 7-14 – Model standu pro výuku servopohonů

### 7.5.1 Simulink PLC Coder

Simulink PLC Coder umožňuje snadný export subsystémů ze Simulinku přímo do .xml souboru. Vytvoří se funkční blok napsaný v syntaxi používaném v TwinCATu. XML je jednoduše nahrán do TwinCAT projektu ve Visual Studiu. Simulink PLC Coder nepodporuje spojité bloky, proto bylo nutné před exportem kaskádní regulaci upravit. K tomu byl použit PI regulátor z knihovny Control System Toolbox. Výsledná podoba kaskádní regulace je vidět níže (Obrázek 7-15):



Obrázek 7-15 – Diskrétní kaskádní regulace

Takto upravený model již bylo možné exportovat pravým kliknutím na subsystém a volbou PLC Coder/Generate Code for Subsystem. Exportovaný model byl nahrán do řídicího projektu pravým kliknutím na složku STAND/PLC/STAND\_PLC/POUs a volbou Import PLCopenXML. Aktualizaci hodnot může zajišťovat program PRG\_SIMULINK\_REG, který načítá v každém taktu NC data z příslušné osy a posílá je do regulátoru. Práce s vygenerovaným funkčním blokem v prostředí TwinCAT je spíše neintuitivní a vzdálená jednoduchým úpravám regulace v Simulinku, což komplikuje vývoj regulátoru. Po odladění řídicího algoritmu se ovšem může jednat o vhodnou metodu implementace, protože funkční bloky jsou snadno přenositelné mezi systémy a nevyžadují licenci nebo instalaci dodatečného softwaru.



## 7.5.2 TwinCAT 3 Target for Simulink

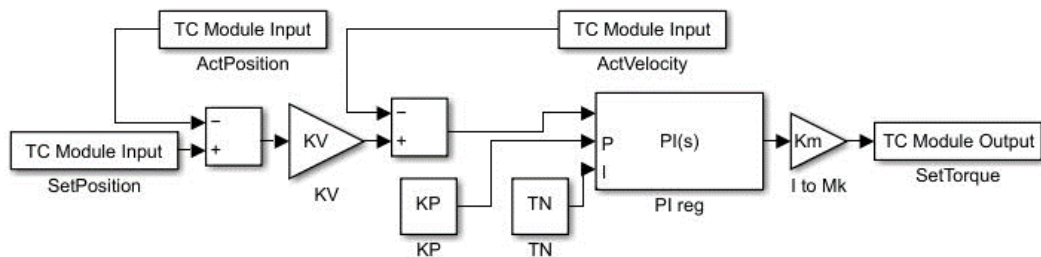
Jestliže bylo zprovoznění Simulink PLC Coderu jednoduché a stačilo mít pouze nainstalovaný příslušný toolbox, u TwinCAT 3 Target for Simulink je tomu zcela opačně. Na notebooku DELL musí být nainstalován Matlab včetně Simulinku a Simulink Coderu, Visual Studio 2017 včetně knihovny Desktop development with C++, TwinCAT 3 a TwinCAT Tools for MATLAB and Simulink. Po instalaci softwaru byl spuštěn Matlab s administrátorskými právy a zadán příkaz pro instalaci patřičných knihoven Target for Simulink: C:\TwinCAT\Functions\TE14xx-ToolsForMatlabAndSimulink\SetupTE14xx.p.

Exportované modely ze Simulinku je nutné podepsat certifikátem, jinak je není možné v TwinCATu spustit. Certifikáty vydává Beckhoff, pro testovací účely je však lze vygenerovat bez oficiálního potvrzení a podepsat jimi vytvořený TcCOM modul (Simulink model). Nejprve je nutné přepnout Windows do testovacího režimu zadáním příkazu „bcdedit /set testsigning yes“ do CMD jako administrátor. Zároveň je potřeba vypnout SecureBoot v BIOSu. V dalším kroku je možné certifikát vygenerovat prostřednictvím Visual Studia, záložky TwinCAT/Software Protection/OEM Certificate/Create New. Certifikát musí obsahovat podpis pro spouštění C++ programů a je vytvořen včetně přístupového hesla. Vygenerovaný certifikát je přiřazen k „Windows environment variable“ s názvem „TcSignTwinCatCertName.“ Přiřazení je možné učinit ve Windows v záložce Systém/Upřesnit nastavení systému/Proměnné prostředí/Nová. Hodnota proměnné odpovídá adrese certifikátu.

Vytvořený OEM certifikát je zapsán do systémových registrů spuštěním souboru C:\TwinCAT\3.1\Target\OemCertificates\NazevOrganizace.reg. Posledním krokem k umožnění podepisování exportovaných modulů je nahrání hesla k certifikátu. V příkazové řádce je spuštěn program TcSignTool.exe na adrese C:\TwinCAT\3.1\sdk\Bin\ TcSignTool.exe. Heslo je přiřazeno příkazem:

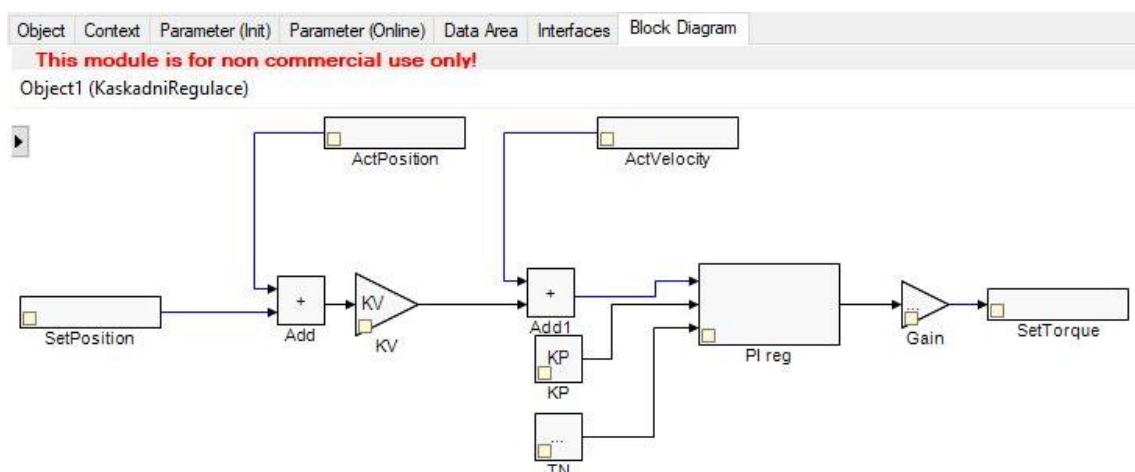
```
tcsigntool grant /f "C:\TwinCAT\3.1\CustomConfig\Certificates\Certifikat.tccert" /p Heslo
```

Od této chvíle jsou všechny vygenerované modely automaticky podepisovány OEM klíčem.



Obrázek 7-16 – Kaskádní regulace upravená pro Target for Simulink

Model regulace v Simulinku byl upraven, vstupy a výstupy subsystému byly nahrazeny bloky TC Model Input a TC Model Output (Obrázek 7-16). Solver byl nastaven na fixní krok s taktom odpovídajícím taktu PLC úlohy. V Simulinku byla spuštěna aplikace Simulink Coder. V nastavení byl vybrán System target file: TwinCATGrt.tcl a v záložce Code Generation/Optimization byla zvolena volba Tunable, čímž byla umožněna změna parametrů regulátoru v prostředí TwinCAT. V tuto chvíli bylo možné přistoupit ke generování kódu kliknutím na ikonu „Generate Code.“ Vygenerovaný objekt byl načten ve Visual Studiu pod záložkou STAND/SYSTEM/TcCOM Objects/Přidat novou položku. K objektu byl přiřazený vytvořený PLC task s názvem Task\_Simulink, který určuje takt, ve kterém je model volán. Na vstupy a výstupy modelu byly namapovány globální proměnné z řídicího projektu: IrActPos, IrActVelo, IrSetPos a IrSetTorque. Regulační konstanty lze upravovat v záložce Block Diagram (Obrázek 7-17) kliknutím na žlutou ikonu u bloku:

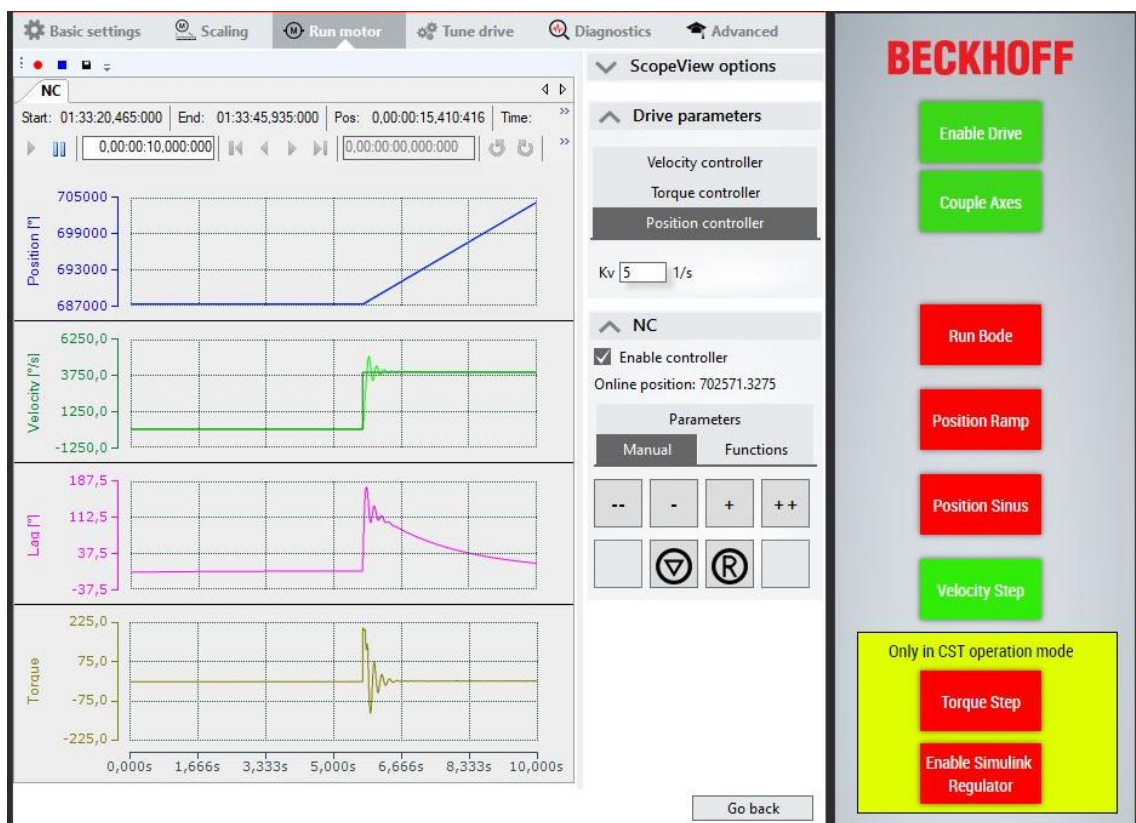


Obrázek 7-17 – Importovaný model regulace do prostředí TwinCAT

Práce s importovným modelem kaskádní regulace pomocí TwinCAT 3 Target for Simulink je jednodušší a názornější než práce s funkčním blokem ze Simulink PLC Coderu. Nevýhodou je zdlouhavá instalace a nepřenositelnost takového modelu na jiné systémy bez platné OEM licence. Stand s propojeným vývojovým prostředím je připraven pro vývoj nových řídicích algoritmů v Simulinku, které je možné nasadit do řídicího systému v řádu minut. Implementovaná regulace byla úspěšně naladěna s parametry, které byly horší než dosažené výsledky s regulací Beckhoff v CSP režimu. V porovnání se základním nastavením regulace se ovšem jedná o slibné výsledky, především s přihlédnutím k použitému taktu komunikace 1 ms. Modely v Simulinku včetně jejich parametrů jsou součástí přílohy MATLAB.zip.

## 7.6 HMI projekt

Stand pro výuku servopohonů má demonstrovat interakci mechanické struktury se servopohony, která se nejlépe projeví při ladění regulace. V TwinCAT 3 probíhá nastavení pohonů v rámci projektu Drive Manager 2. Tomu bylo podřízeno i uživatelské rozhraní HMI (Obrázek 7-18), které si klade za cíl toto prostředí doplnit běžně užívanými

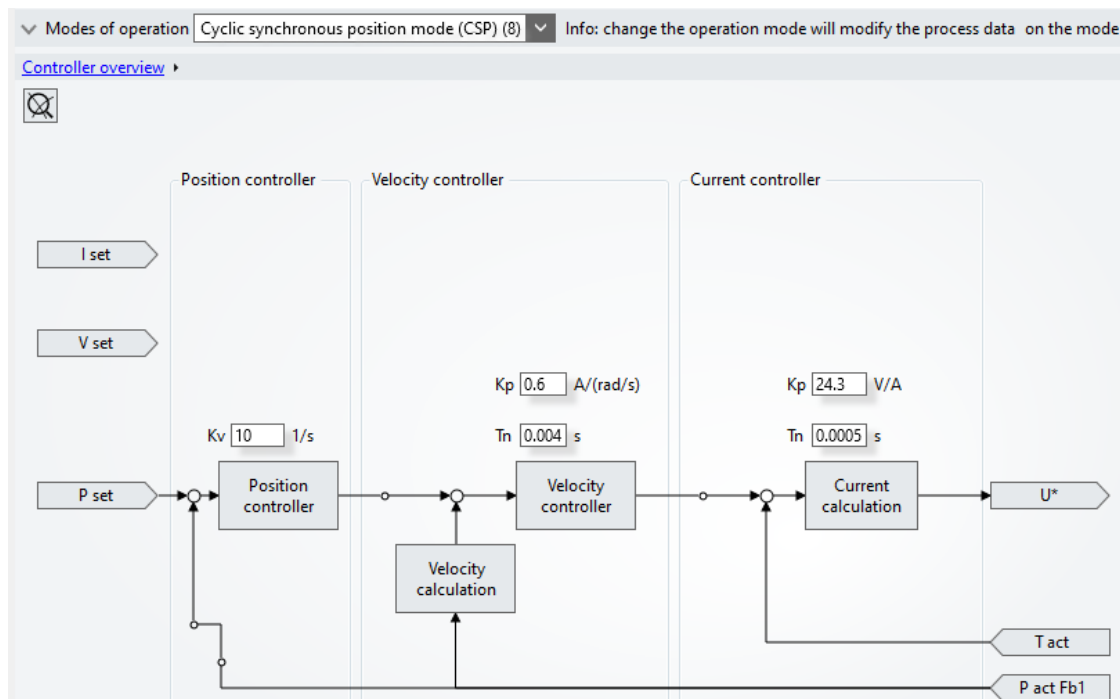


Obrázek 7-18 – Drive Manager 2 a HMI

funkcemi. Tím zjednodušuje obsluhu standu, aniž by utrpěla odbornost předaných informací. Uživatelské rozhraní bylo navrženo jako úzký pruh na pravé straně obrazovky, na levé straně se nachází karta „Run motor“ v Drive Manager 2 pro ladění regulace.

Pohony se spouštějí tlačítkem Enable Drive na pravé straně. Pokud je pohon připravený, tlačítko se zbarví zeleně, v opačném případě zůstává červené. Couple Axes zajistí couplování os, níže je volba spuštění měření Bode diagramu Run Bode a pohyby os Position Ramp, Position Sinus a Velocity Step. Poslední dvě tlačítka HMI panelu lze použít pouze pokud je pohon v momentovém režimu CST. Tlačítko Torque Step spustí skok momentu, Enable Simulink Regulator aktivuje kaskádní regulaci importovanou z vývojového prostředí Simulink. Na levé straně se v Drive Manageru nacházejí tlačítka pro polohování pohonu (+ a -) a pro rychloposuv (++ a --). Záložka Functions nabízí další možnosti pohybů. Uživatelský manuál je součástí textové přílohy.

Regulační konstanty jsou dostupné v prostřední sekci Drive parameters. Levá strana zobrazuje průběh aktuální polohy, rychlosti, polohové odchylky a krouticího momentu. Zobrazení lze upravovat pod kartou ScopeView options, případně všechny průběhy exportovat do samostatného Scope projektu. Struktura kaskádní regulace Beckhoff je patrná v záložce Tune Drive (Obrázek 7-19).



Obrázek 7-19 – Schéma regulace a přepínání režimů

V horní části je možné přepínat mezi režimy polohového řízení CSP, rychlostního řízení CSV nebo momentového řízení CST. Podle toho se interaktivně mění zapojení na schématu níže. Pro ladění regulace pouze jednoho motoru je nejjednodušší nastavit na druhém motoru režim CST, oba motory aktivovat a následně pokračovat v ladění například Ziegler-Nicholsonovou metodou. Stejný postup lze opakovat i pro druhý motor. Uživatelské rozhraní HMI bylo vytvořeno s využitím modulu TE2000 TwinCAT 3 HMI Engineering. Na notebooku DELL běží paralelně s TwinCAT XAR také HMI server, který se stará o zobrazování dat a obsluhu PLC vstupů a výstupů propojených s HMI projektem.

## 7.7 Realizace výukového standu

Řídicí projekt byl testován na standu, který byl dle návrhu nad rámec požadavků zadání vyroben a sestaven (Obrázek 7-20). Na základě praktické zkušenosti při sestavování pracoviště byly odladěny rozměry modelu a výkresů některých dílců. Příloha již obsahuje takovou dokumentaci, která jde bez problémů smontovat. Součástí přílohy je 3D model pracoviště DP1835-01-00-STAND-PRO-VYUKU-SERVOPOHONU.stp a sestavný výkres s rozpiskou všech dílů DP1835-01-00. Pro výrobu setrvačníku je v příloze výrobní výkres DP1835-01-01. Polykarbonátové krytování bylo frézováno na plotru, plexisklové kryty byly vypalovány laserem. Soubory DP1835-01-02 až DP1835-01-08 ve formátu .dwg pro výrobu jsou rovněž přiloženy. Krytky setrvačnicků, zasklívací profil, držák zdroje a držák DIN lišty byly vytisknuty na 3D tiskárně z materiálu PETG, v příloze jsou .stl soubory pro tisk a .step soubory pro případnou úpravu modelů. Základní zapojení elektroinstalace je patrné z blokového schéma DP1835-Blokove-schema-elektroinstalace.pdf.

Archiv řídicího projektu je součástí přílohy DP1835-TWINCAT-PROJEKT.zip spolu s exportovanými vytvořenými zdrojovými kódy ve formátu .pdf. Po instalaci všech softwarových modulů je možné do Visual Studia 17 projekt snadno nahrát a dle uživatelského manuálu v příloze zprovoznit.



*Obrázek 7-20 – Stand pro výuku servopohonů včetně řídicího notebooku*

## 8 Závěr

Cílem práce bylo navrhnout stand pro výuku servopohonů, který by demonstroval interakci pohonů s mechanickou strukturou. Základem takového pracoviště je řídicí systém, proto byla v úvodu práce provedena rešerše operačních systémů reálného času. Uvedena byla řešení od společností Beckhoff, MathWorks, REXYGEN a National Instruments. Vývojová prostředí těchto systémů byla vzájemně porovnána, řešila se rovněž jejich oboustranná kompatibilita v případě kombinace. Pro každé řešení byly představeny možnosti řízení pohonů. Ve druhé části rešerše byly uvedeny komerčně dostupné standy a pracoviště z odborných publikací.

Z rešeršní části vyplynula absence dostupného výukového standu na průmyslovém standartu. Proto bylo v praktické části přistoupeno k návrhu pracoviště. Nejprve byly představeny tři konstrukční varianty s různými řídicími systémy, které byly vzájemně porovnány. Vybrán byl stand na platformě Beckhoff tvořený dvojicí motorů, propojených poddajnou hřídelí. Setrvačnost každého motoru byla zvýšena přídatným setrvačником. Jedním z požadavků byla nízká vlastní frekvence pohonu tak, aby se při řízení negativně projevil jeho mechanické vlastnosti. Ve výpočtové části byla vybrána vhodná velikost pohonu, pro který byly stanoveny rozměry komponent. Propojení motorů a poddajné hřídele je zajištěno lamelovou spojkou, na kterou je svěrně uložen setrvačnik. Pohon je umístěn na hliníkové desce s hliníkovým rámem a polykarbonátovým krytváním. Prostor je rozdělen na elektronickou a mechanickou část. V elektronické části se nacházejí měniče EL7221 a coupler EK1100, napájecí zdroje a bezpečnostní relé AD SRE4C propojené s tlačítkem nouzového zastavení. Chlazení zajišťuje dvojice ventilátorů a termostat.

Druhá polovina praktické části se zabývala návrhem řídicího projektu TwinCAT. Nejprve bylo představeno základní nastavení systému, následoval popis struktury řídicího projektu. Pro usnadnění ovládání pracoviště byly naprogramovány funkční bloky. Jednalo se o bloky pro zapnutí a vypnutí pohonů, pohyb po zvolené trajektorii a blok pro umožnění měření Bode diagramu včetně exportu dat ve formátu vhodném pro Matlab. Do řídicího projektu byla úspěšně implementována kaskádní regulace ze Simulinku. Simulink PLC Coder je vhodný nástroj pro implementaci regulace do cílových zařízení formou funkčního bloku, protože vygenerovaný kód již nevyžaduje další licence.



Modul TwinCAT 3 Target for Simulink se ukázal být jako účinný prostředek pro vývoj. Stand s takto propojeným vývojovým prostředím je připraven pro tvorbu nových řídicích algoritmů v Simulinku, které je možné nasadit do TwinCATu v řádu minut. V poslední podkapitole byl popsán vytvořený HMI projekt doplňující nástroj Drive Manager 2. Kombinace těchto prostředí nabízí pohodlné ladění kaskádní regulace včetně sledování aktuálních hodnot polohy, rychlosti, polohové odchylky a krouticího momentu.

Stand pro výuku servopohonů byl dle návrhu nad rámec požadavků zadání vyroben a sestaven. Na základě praktické zkušenosti při sestavování pracoviště byly odladěny rozměry modelu a výkresů některých dílců. Řídicí systém byl implementován prostřednictvím notebooku DELL včetně Matlabu a všech jeho součástí. Propojení se standem bylo realizováno ethernetovým kabelem. Dokončené pracoviště je připravené pro nasazení ve výuce servopohonů s důrazem na průmyslové aplikace.



# Seznamy

## Seznam použité literatury

- [1] BECKHOFF AUTOMATION GmbH & Co. KG. Beckhoff Information System [online]. 2023 [cit. 2022-10-18]. Dostupné z: [https://infosys.beckhoff.com/english.php?content=../content/1033/te1010\\_tc3\\_realtime\\_monitor/45036003102573963.html&id=](https://infosys.beckhoff.com/english.php?content=../content/1033/te1010_tc3_realtime_monitor/45036003102573963.html&id=)
- [2] HAMBARDE, Prasana; VARMA, Rachit; JHA, Shivani. The Survey of Real Time Operating System: RTOS. 2014 International Conference on Electronic Systems, Signal Processing and Computing Technologies, 2014. ISBN 978-1-4799-2102-7.
- [3] SAGATOWSKI, Jakob. TwinCAT 3 Course. 2021 [cit. 2022-10-29]. Dostupné z: <https://alltwincat.com/twincat-3-tutorial/>
- [4] BECKHOFF AUTOMATION GmbH & Co. KG. TwinCAT automation software [online]. [cit. 2023-1-22]. Dostupné z: <https://www.beckhoff.com/cs-cz/products/automation/twincat/>
- [5] DI PIETRO, Roberto; LOMBARDI, Flavio. Virtualization Technologies and Cloud Security: advantages, issues, and perspectives. College of Science and Engineering, Hamad Bin Khalifa University - Doha, Qatar [online]. 2018 [cit. 2022-11-2]. Dostupné z: [https://www.researchgate.net/publication/326696873\\_Virtualization\\_Technologies\\_and\\_Cloud\\_Security\\_advantages\\_issues\\_and\\_perspectives](https://www.researchgate.net/publication/326696873_Virtualization_Technologies_and_Cloud_Security_advantages_issues_and_perspectives)
- [6] SPEEDGOAT GmbH. Real-Time Target Machines [online]. [cit. 2023-1-22]. Dostupné z: <https://www.speedgoat.com/products-services/real-time-target-machines>
- [7] THE MATHWORKS, Inc. Simulink Real-Time [online]. [cit. 2022-11-5]. Dostupné z: <https://uk.mathworks.com/products/simulink-real-time.html>
- [8] THE MATHWORKS, Inc. Simulink Desktop Real-Time [online]. [cit. 2022-11-5]. Dostupné z: <https://uk.mathworks.com/products/simulink-desktop-real-time.html>
- [9] REX CONTROLS s.r.o. REXYGEN – Přehled podporovaných zařízení [online]. [cit. 2022-12-3]. Dostupné z: <https://www.rexygen.com/cs/>

- [10] NATIONAL INSTRUMENTS CORP. Building a Real-Time System With NI Hardware and Software [online]. 2023 [cit. 2022-11-26]. Dostupné z: <https://www.ni.com/cs-cz/innovations/white-papers/06/building-a-real-time-system-with-ni-hardware-and-software.html>
- [11] NATIONAL INSTRUMENTS CORP. Introduction to NI Linux Real-Time [online]. 2023 [cit. 2022-11-26]. Dostupné z: <https://www.ni.com/cs-cz/shop/linux/introduction-to-ni-linux-real-time.html>
- [12] NATIONAL INSTRUMENTS CORP. PXI Systems [online]. 2023 [cit. 2023-1-4]. Dostupné z: <https://www.ni.com/cs-cz/shop/pxi.html>
- [13] BECKHOFF AUTOMATION GmbH & Co. KG. TE1400 TC3 SIMULINK [online]. [cit. 2023-1-4]. Dostupné z: <https://www.beckhoff.com/en-en/products/automation/twincat/texxxx-twincat-3-engineering/te1400.html>
- [14] THE MATHWORKS, Inc. Simulink PLC Coder [online]. [cit. 2023-1-4]. Dostupné z: <https://uk.mathworks.com/products/simulink-plc-coder.html>
- [15] REX CONTROLS s.r.o. Driver for communication with Siemens PLCs. Plzeň, Česká republika [online]. [cit. 2023-1-5]. Dostupné z: [https://www.rexygen.com/doc/ENGLISH/MANUALS/S7Drv/S7Drv\\_ENG.html#x1-30001.1](https://www.rexygen.com/doc/ENGLISH/MANUALS/S7Drv/S7Drv_ENG.html#x1-30001.1)
- [16] VRÁNA, Stanislav. Vyhodnocování signálů při samoseřizování s využitím spolupráce programů REX a MATLAB. České vysoké učení technické v Praze, Praha, Česká republika [online]. 2006 [cit. 2023-1-5]. Dostupné z: <https://www2.humusoft.cz/www/papers/tcp07/vrana.pdf>
- [17] BECKHOFF AUTOMATION GmbH & Co. KG. Beckhoff TF5xxx – Motion [online]. [cit. 2023-1-22]. Dostupné z: [https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib\\_tc2\\_mc2/70091915.html&id=](https://infosys.beckhoff.com/english.php?content=../content/1033/tcplclib_tc2_mc2/70091915.html&id=)
- [18] NATIONAL INSTRUMENTS CORP. Using the NI SoftMotion Axis Interface [online]. 2009 [cit. 2022-11-26]. Dostupné z: <https://download.ni.com/support/manuals/372597a.pdf>

- [19] NATIONAL INSTRUMENTS CORP. SoftMotion Compatibility with LabVIEW Real Time Controllers [online]. 2023 [cit. 2023-1-20]. Dostupné z: <https://www.ni.com/cs-cz/support/documentation/compatibility/17/softmotion-compatibility-with-labview-real-time-controllers.html>
- [20] DANĚK, Martin. Programovatelná hradlová pole – FPGA [online]. Ústav teorie informace a automatizace Akademie věd ČR, Praha, Česká republika. Dostupné z: [https://automa.cz/cz/casopis-clanky/programovatelnahradlova-pole-fpga-2006\\_02\\_30930\\_672/s](https://automa.cz/cz/casopis-clanky/programovatelnahradlova-pole-fpga-2006_02_30930_672/s)
- [21] NATIONAL INSTRUMENTS CORP. Getting Started with LabVIEW FPGA [online]. 2023 [cit. 2023-1-20]. Dostupné z: <https://learn.ni.com/learn/article/getting-started-with-fpga#toc1>
- [22] THE MATHWORKS, Inc. HDL Coder [online]. [cit. 2023-1-20]. Dostupné z: <https://uk.mathworks.com/products/hdl-coder.html>
- [23] EDIBON. Servosystems Trainer (DC motors) SERIN/CCB [online]. 2009 [cit. 2023-1-6] Dostupné z: <https://www.edibon.com/en/servosystems-basic-unit-for-dc-motors/catalog>
- [24] HUNT ENGINEERING. Choosing FPGA or DSP for your Application [online]. 2010 [cit. 2023-1-6]. Dostupné z: <http://hunteng.co.uk/info/fpga-or-dsp.htm>
- [25] GOOGOL TECHNOLOGY Co., Ltd. DC Servo Control Trainer [online]. 2020 [cit. 2022-10-17]. Dostupné z: [http://www.googoltech.com/pro\\_view-68.html](http://www.googoltech.com/pro_view-68.html)
- [26] GOUBEJ, Martin; BLÁHA, Lukáš. Raspberry Pi-based Motion Control Testbed for Mechatronics Education. NTIS Research Centre, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Czechia, 2022. ISSN 2405-8963.
- [27] MAXON INTERNATIONAL Ltd. MAXPOS 50/5, digital positioning controller, 5 A, 10 - 50 VDC [online]. 2017 [cit. 2023-1-7]. Dostupné z: <https://www.maxongroup.com/maxon/view/product/control/Positionierung/447293>
- [28] REX CONTROLS s.r.o. Monarco HAT Hardware Reference Manual [online]. 2022 [cit. 2023-7-1]. Dostupné z: <https://www.monarco.io/docs/Monarco-HAT-Hardware-Reference-Manual.pdf>

- [29] HOFFMANN, Nils; THOMSEN, Sönke; FUCHS, Friedrich. Model Based Predictive Speed Control of a Drive System with Torsional Loads – A Practical Approach [online]. 14th International Power Electronics and Motion Control Conference, EPE-PEMC 2010. DOI: 10.1109/EPEPEMC.2010.5606693.
- [30] SAEY, Philippe; CAUWE, X; DECONINCK, Geert. Drive systems with torsional load: versatile low-cost educational laboratory set-up [online]. Proceedings of the 2011 14th European Conference on Power Electronics and Applications, Birmingham, UK, 2011. ISBN: 978-90-75815-14-6.
- [31] THE MATHWORKS, Inc. xPC Target [online]. 2005 [cit. 2023-1-7]. Dostupné z: [http://www.bmed.mcgill.ca/reklab/archive/xPC/xPC\\_documentation/xpc\\_target\\_ug\[1\].pdf](http://www.bmed.mcgill.ca/reklab/archive/xPC/xPC_documentation/xpc_target_ug[1].pdf)
- [32] BECKHOFF AUTOMATION GmbH & Co. AM8100 Operating instructions [online]. [cit. 2023-5-20]. Dostupné z: [https://download.beckhoff.com/download/Document/motion/am8100\\_ba\\_en.pdf](https://download.beckhoff.com/download/Document/motion/am8100_ba_en.pdf)
- [33] KTR CR, spol s.r.o. Drive Technology katalog. [cit. 2023-6-21]. Dostupné z: [https://www.ktr.com//catalog/index.php?catalog=DriveTechnology#page\\_154](https://www.ktr.com//catalog/index.php?catalog=DriveTechnology#page_154)
- [34] Haberkorn s.r.o. Korýtková spojka svěrná. [cit. 2023-6-20]. Dostupné z: <https://eshop.haberkorn.cz/korytkova-spojka-sverna-1.html>
- [35] SOUČEK, Pavel. Servomechanismy ve výrobních strojích. Vydavatelství ČVUT, 2004. ISBN 80-01-02902-6
- [36] SOUČEK, Pavel. Vložené převody v pohonech NC strojů a robotů. 2019.

## Seznam obrázků

Obrázek 3-1 - Nedeterministické chování běžného operačního systému [3].....	13
Obrázek 3-2 – Operační systém reálného času [3] .....	14
Obrázek 3-3 - TwinCAT 3 [4].....	15
Obrázek 3-4 - RTOS na izolovaném jádru procesoru [1] .....	16
Obrázek 3-5 - RTOS na jádře spolu s OS [1] .....	16
Obrázek 3-6 – Okruhy privilegií pro procesory s architekturou x86 [5].....	18
Obrázek 3-7 – Simulink Desktop Real-Time v externím módu [8].....	18
Obrázek 3-8 Struktura ADS komunikace [1].....	25
Obrázek 3-9 Propojení modulu NI SoftMotion [18].....	29
Obrázek 3-10 -Výuková stanice Edibon SERIN/CA [23].....	31
Obrázek 3-11 - GSMT2014 [24].....	33
Obrázek 3-12 - Flexible two-mass motion systém [26].....	34
Obrázek 3-13 – Schéma testovacího standu a torzní model soustavy [29].....	36
Obrázek 3-14 - Motion Control Demonstrator [30].....	38
Obrázek 5-1 Možnosti rozmístění komponent .....	40
Obrázek 5-2 – Varianta 1.....	41
Obrázek 5-3 – Varianta 2.....	43
Obrázek 5-4 – Varianta 3.....	45
Obrázek 6-1 – Dovolené radiální zatížení motoru AM8131 [33] .....	51
Obrázek 6-2 – Vlnovcová spojka .....	53
Obrázek 6-3 – Svěrná korýtková spojka [35].....	53
Obrázek 6-4 – Lamelová spojka kardanová [36] .....	54
Obrázek 6-5 – Uložení setrvačníku na spojku svěrným pouzdrem .....	55
Obrázek 6-6 – Uložení setrvačníku na spojku svěrně šrouby .....	55
Obrázek 6-7 – Uložení setrvačníku na spojku čelními šrouby.....	56
Obrázek 6-8 – Uložení setrvačníku na spojku čelními šrouby.....	56
Obrázek 6-9 – Uložení setrvačníku se zajištěním stavěcími šrouby.....	56
Obrázek 6-10 – Pohon na hliníkové desce .....	60
Obrázek 6-11 – Rám standu s krytováním a pohonem.....	61
Obrázek 6-12 – Systém chlazení standu .....	62

Obrázek 7-1 – Nastavení Real-Time systému.....	67
Obrázek 7-2 – Základní pohyb NC osy.....	68
Obrázek 7-3 – Stavový diagram řídicího projektu.....	70
Obrázek 7-4 – Drive Manager 2 a vytvořené HMI .....	71
Obrázek 7-5 – Vstupy a výstupy funkčního bloku FB_AXIS_STATUS_CTRL .....	72
Obrázek 7-6 – Vstupy a výstupy funkčního bloku FB_ENABLE_AXIS .....	73
Obrázek 7-7 – Vstupy a výstupy funkčního bloku FB_BODE.....	74
Obrázek 7-8 – Stavový diagram FB_ENABLE, FB_AXIS_STATUS_CTRL, FB_BODE ...	74
Obrázek 7-9 – Příklad zpracování dat do Bodeho diagramu.....	76
Obrázek 7-10 – Vstupy a výstupy funkčního bloku FB_MOVEMENT.....	77
Obrázek 7-11 – Stavový diagram FB_MOVEMENT .....	78
Obrázek 7-12 – Vstupy a výstupy funkčního bloku FB_STEP_TORQUE .....	78
Obrázek 7-13 – Stavový diagram FB_STEP_TORQUE.....	79
Obrázek 7-14 – Model standu pro výuku servopohonů .....	80
Obrázek 7-15 – Diskrétní kaskádní regulace .....	80
Obrázek 7-16 – Kaskádní regulace upravená pro Target for Simulink.....	82
Obrázek 7-17 – Importovaný model regulace do prostředí TwinCAT .....	82
Obrázek 7-18 – Drive Manager 2 a HMI.....	83
Obrázek 7-19 – Schéma regulace a přepínání režimů.....	84
Obrázek 7-20 – Stand pro výuku servopohonů včetně řídicího notebooku .....	86

## Seznam tabulek

Tabulka 5-1 – Porovnání variant .....	48
Tabulka 6-1 Přehled základních konstrukčních parametrů pro vybrané motory ....	52

## Seznam použitého softwaru

- Visual Studio 2017 (15.9.54)
  - Vývoj desktopových aplikací pomocí C++
- Beckhoff TwinCAT 3 (Build 4024.10)
  - TE5950 Drive Manager 2 (V1.1.62.0)
  - TE5950 Drive Manager 1 (V1.1.48.0)

- TE14xx Tools for MATLAB and Simulink (2.7.2.0)
- TE2000 HMI Engineering (1.12.760.37)
- TF3300 TwinCAT 3 Measurement (3.4.3147.9)
- TF4100 TwinCAT 3 Controller Toolbox (3.4.1.4)
- Matlab R2022a
  - Control System Toolbox
  - Simulink PLC Coder
- Microsoft Office
- LibreOffice
- Siemens NX

## Seznam příloh

- Textové přílohy
  - Příloha 1 – Uživatelský manuál
- Výkresová dokumentace
  - Standu pro výuku servopohonů
    - DP1835-01-00 – STAND PRO VÝUKU SERVOPOHONŮ
    - DP1835-01-01 – SETRVAČNÍK
    - DP1835-01-02 – KRYT LEVÝ
- Elektronické přílohy
  - 3D modely
    - DP1835-01-00-STAND-PRO-VYUKU-SERVOPOHONU.stp
    - DP1835-01-09-KRYTKA-SETRVACNIK.stp
    - DP1835-01-10-ZASKLIVACI-PROFIL.stp
    - DP1835-01-11-DRZAK-ZDROJ.stp
    - DP1835-01-12-DRZAK-DIN.stp

- Výrobní dokumentace
  - Polymethylmethakrylát (plexisklo) – řezání laserem
    - DP1835-01-02-KRYT-LEVY.dwg
    - DP1835-01-03-KRYT-PRAVY.dwg
    - DP1835-01-04-KRYT-STRED.dwg
    - DP1835-01-05-KRYT-VRCHNI-ZADNI.dwg
  - Polykarbonát – frézování na plotteru
    - DP1835-01-06-KRYT-VRCHNI-PREDNI.dwg
    - DP1835-01-07-KRYT-CELNI.dwg
    - DP1835-01-08-KRYT-BOCNI.dwg
  - PETG – 3D tisk
    - DP1835-01-09-KRYTKA-SETRVACNIK.stl
    - DP1835-01-10-ZASKLIVACI-PROFIL.stl
    - DP1835-01-11-DRZAK-ZDROJ.stl
    - DP1835-01-12-DRZAK-DIN.stl
- Řídicí projekt TwinCAT (DP1835-TWINCAT-PROJEKT.zip)
  - STAND.tnzip (TwinCAT 3 archiv)
  - PDFzdrojoveKodyTwinCAT.zip (Kódy ve formátu .pdf)
- Skripty a modely z Matlabu a Simulinku (DP1835-MATLAB.zip)
  - Návrh konstrukce standu a kontrolní výpočty
    - Konstrukcni\_parametry\_AM8121.m (Matlab)
    - Konstrukcni\_parametry\_AM8131.m (Matlab)
    - Konstrukcni\_parametry\_AM8141.m (Matlab)
    - Kontrolni\_vypocty\_pro\_motor\_AM8131.m (Matlab)
  - Model standu v Simulinku a parametry modelu
    - ModelStandProVyukuServopohonu.slx (Simulink)
    - ParametryKaskadniRegulace.m (Matlab)
  - Export modelu pomocí Simulink PLC Coder
    - KaskadniRegulaceDiskretni.slx (Simulink)
    - KaskadniRegulaceDiskretni.st (TwinCAT 3)
    - KaskadniRegulaceDiskretni.xml (TwinCAT 3)

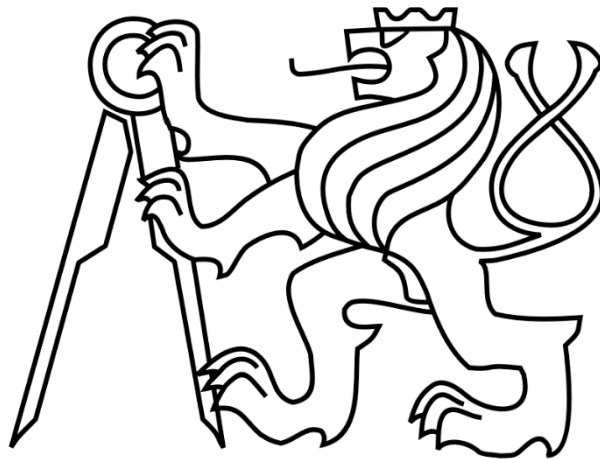


- Export modelu pomocí TwinCAT 3 Target for Simulink
  - KaskadniRegulace.slx (Simulink)
  - ParametryKaskadniRegulaceTarget.m (Matlab)
  - KaskadniRegulace.zip (TwinCAT 3)
- Skript pro výpočet diskrétního přenosu numerické derivace
  - Diskretni\_derivace\_Beckhoff.m (Matlab)
- Skript pro zpracování naměřených dat pro Bode diagram
  - Bode\_zpracovani\_dat.m (Matlab)
  - Bode\_zpracovani\_dat\_linreg.m (Matlab)
- Elektrické zapojení standu
  - DP1835-Blokove-schema-elektroinstalace.pdf
- Soupis nakupovaných komponent
  - DP1835-Soupis-nakupovanych-komponent.xlsx

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ

Ústav výrobních strojů a zařízení



# Diplomová práce

Stand pro výuku servopohonů

Textové přílohy

*Bc. Natan Ber*

# Příloha 1 – Uživatelský manuál

## 1 Zapnutí standu a spuštění pohonů

Před spuštěním zapojte stand do zásuvky a ujistěte se, že není zmáčknuté tlačítko pro nouzové zastavení. Připojte ethernetový kabel a přepněte vypínač do polohy ON:



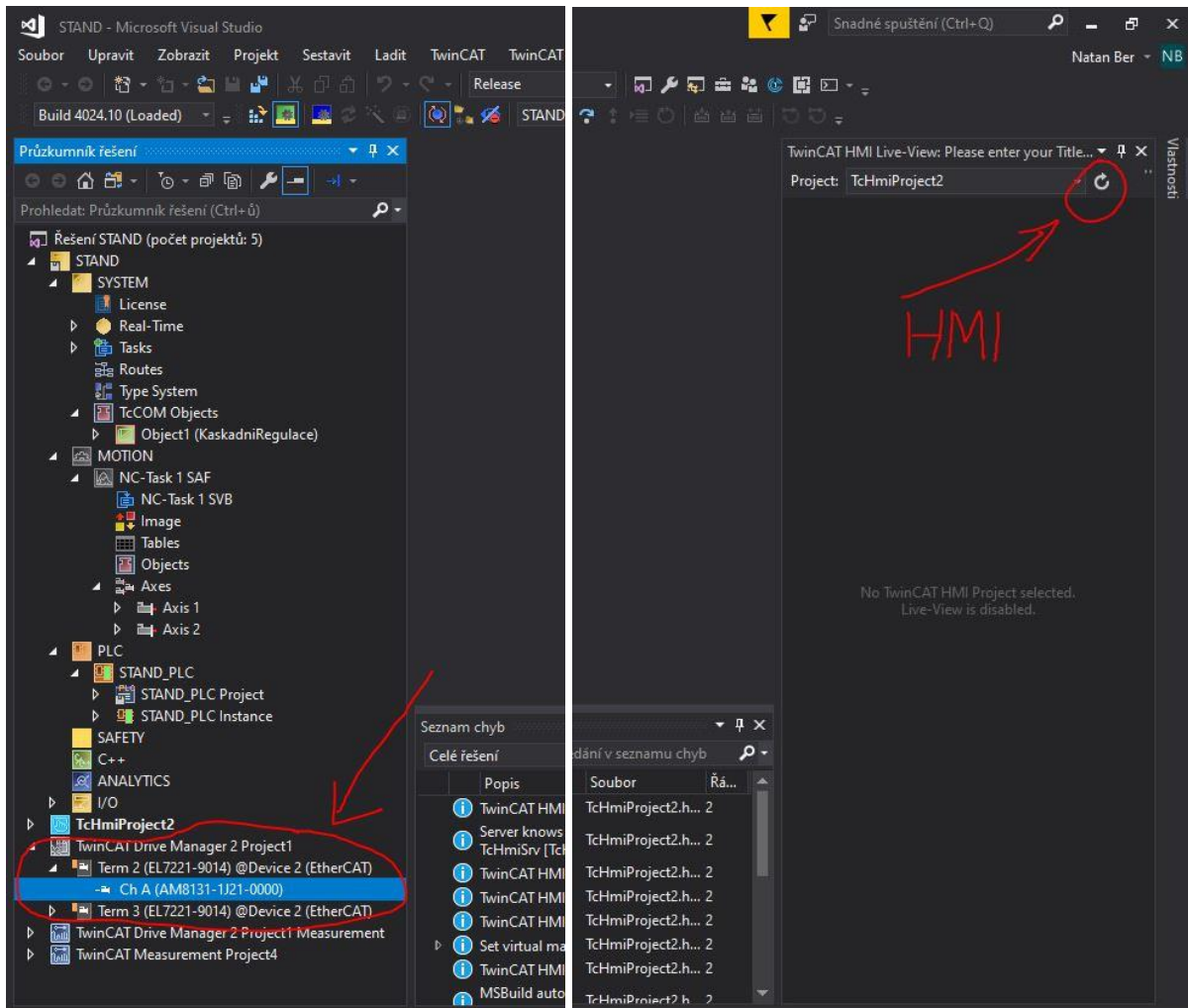
Zapojte ethernetový kabel do notebooku DELL:



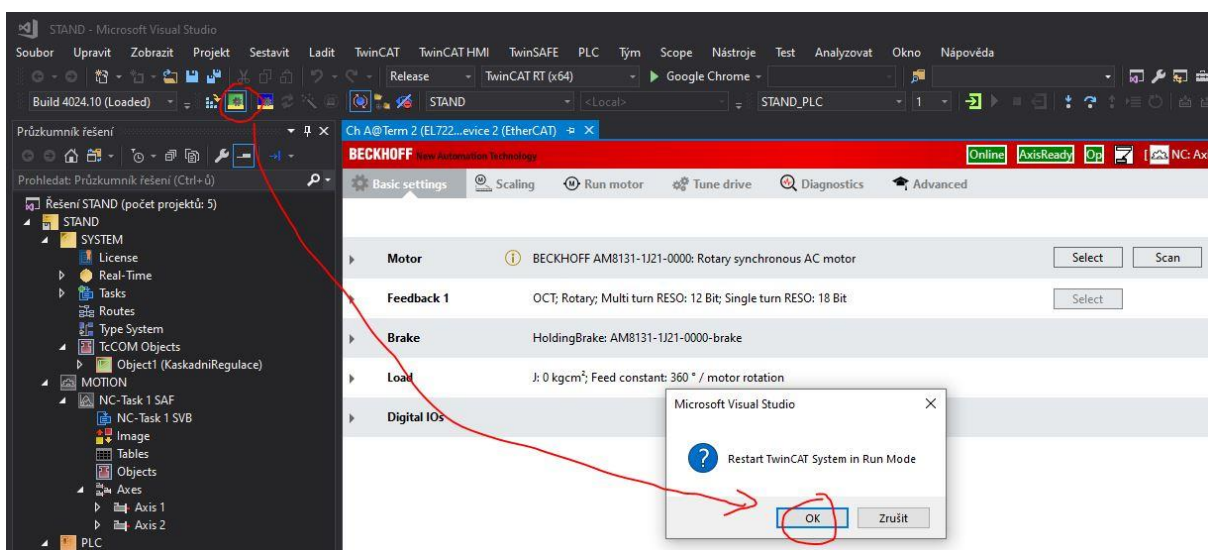
Zapněte notebook a spusťte TwinCAT 3 projekt uložený na ploše ve složce STAND pod názvem STAND.sln:



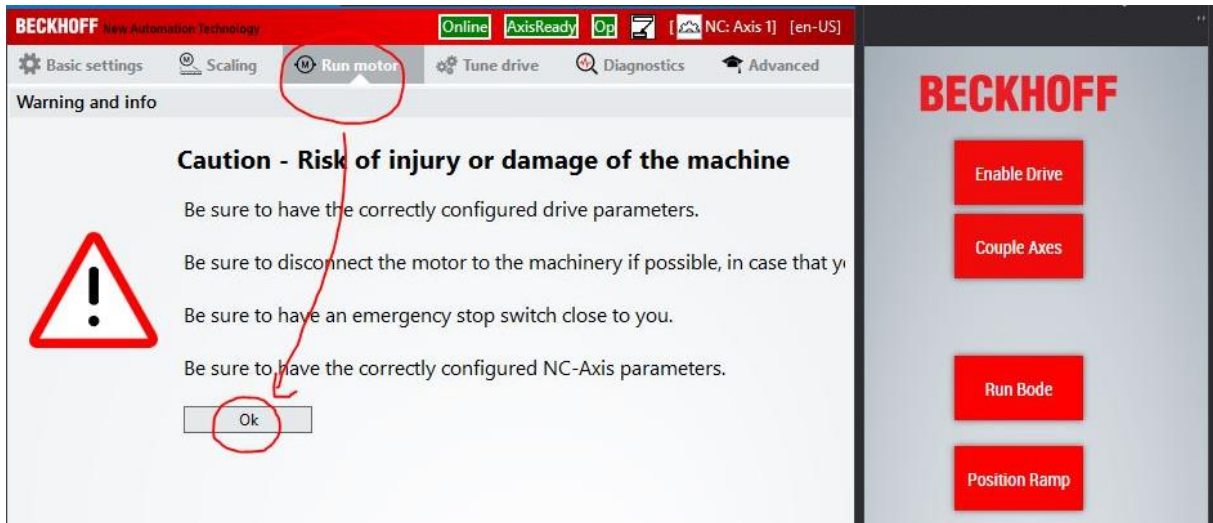
Po načtení projektu ve Visual Studio 17 spusťte HMI kliknutím na tlačítko „refresh“ na pravé straně a Drive Manager 2 volbou „Ch A (AM8131-1J21-0000)“ na pravé straně:



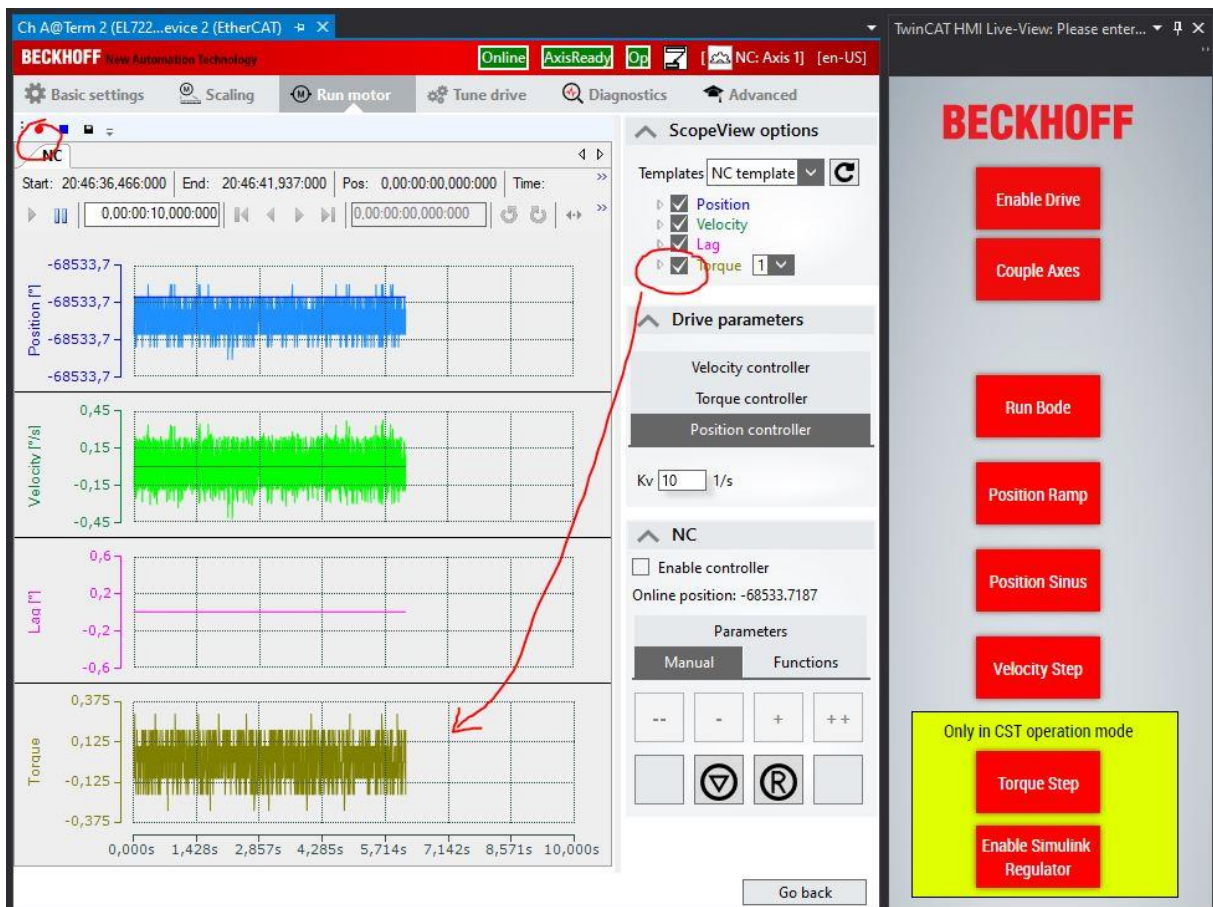
Spusťte PLC přejitím do režimu „Run“ kliknutím na zelenou ikonu na horním panelu a potvrzením vyskakovacího okna. V případě potřeby opište CAPTCHA kód:



V Drive Manager 2 přejděte na záložku „Run motor“ a potvrďte tlačítkem „Ok“:

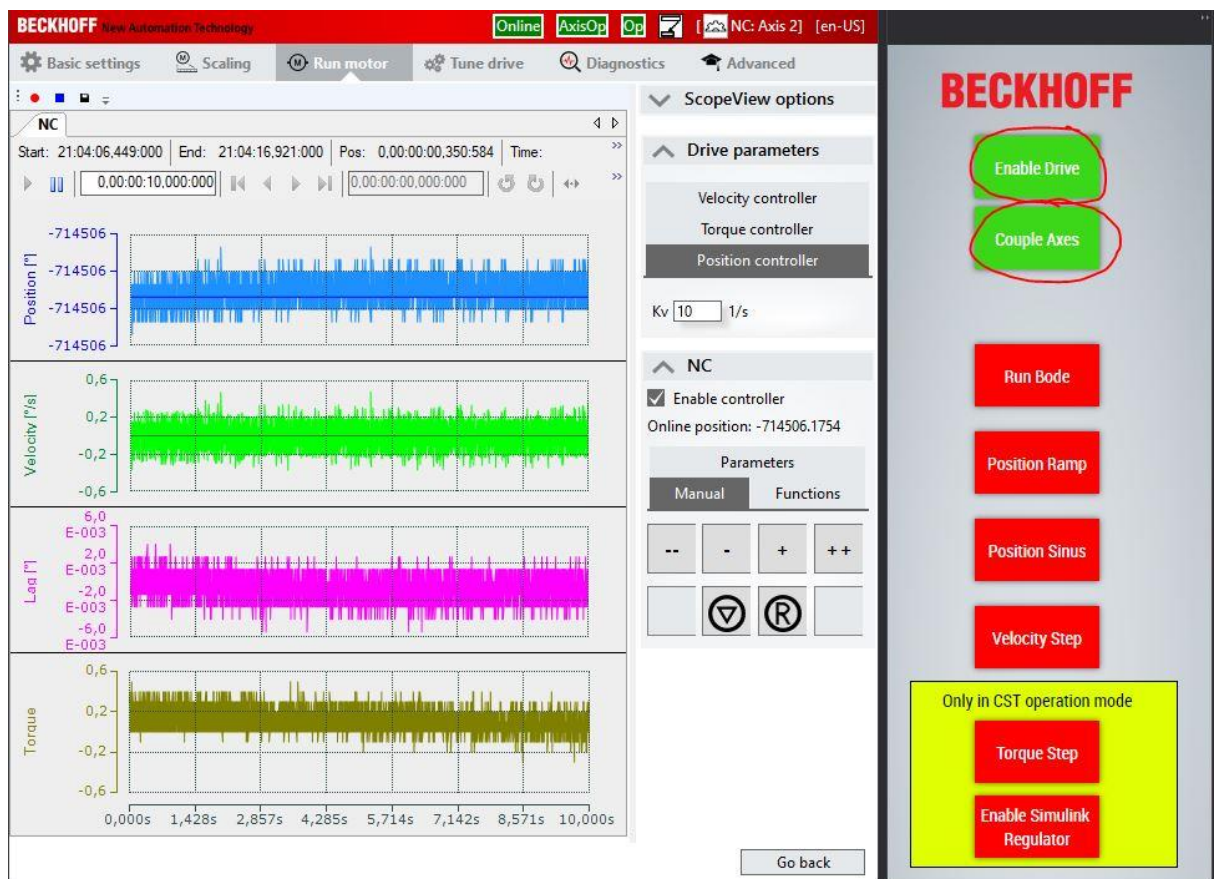


V záložce „ScopeView options“ je možné zobrazit vedle polohy, rychlosti a polohové odchylky také velikost krouticího momentu. Měření se spustí stisknutím červeného tlačítka na levé straně:





Pro zapnutí pohonů stiskněte tlačítko „Enable drive“. Motory jsou po spuštění standartně couplování, je možné je decoulovat a znovu couplovat tlačítkem „Couple Axes.“ Pokud jsou obě osy v režimu CSP nebo CSV, nejsou couplování a pro jednu osu je zadán požadavek pro pohyb, dojde k vypnutí pohonů z důvodu proudové ochrany. Motory mají rozdílné setpointy polohy a tlačí proti sobě, čímž narůstá proud až do kritické hodnoty, při které dojde k chybě a vypnutí pohonů. Při stisknutí tlačítka „Couple Axes“ dojde vždy ke spuštění pohonů:

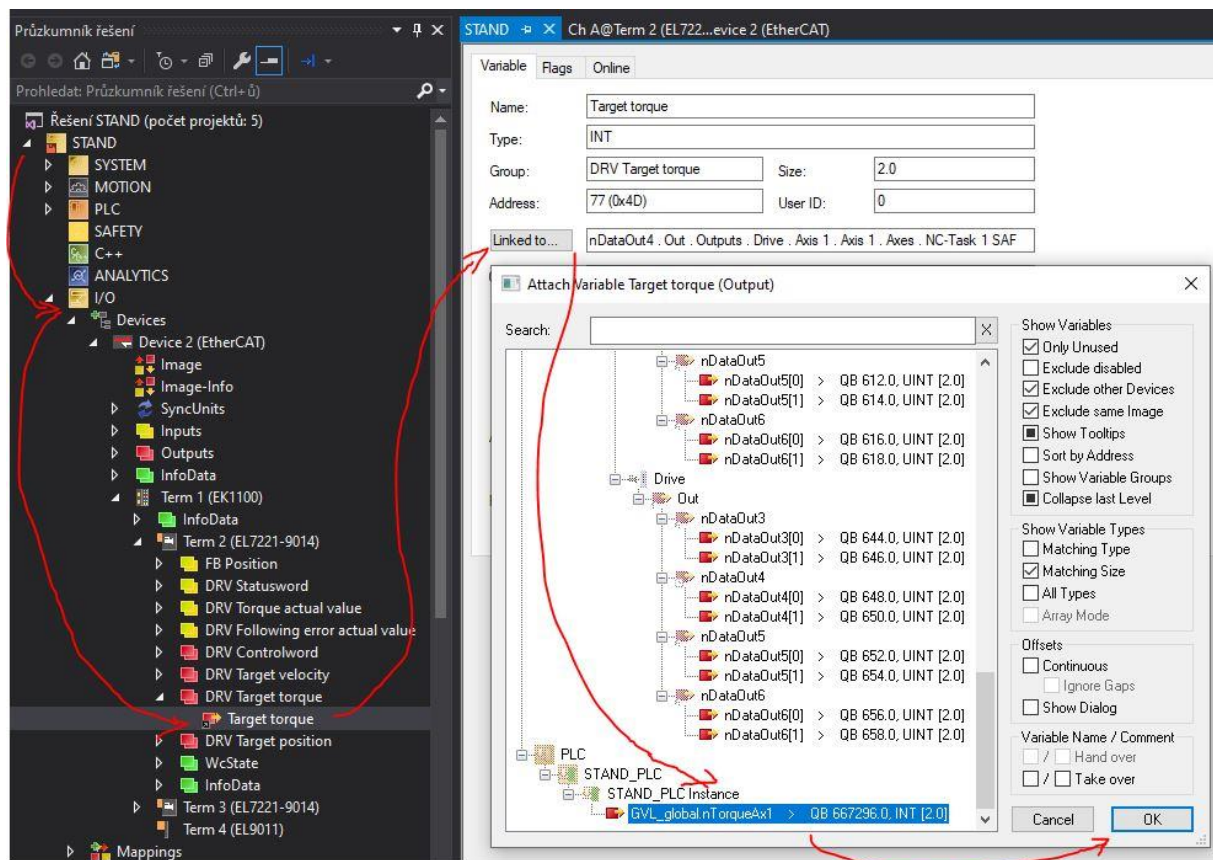


## 2 Přepínání režimů řízení

V Drive Manager 2 v záložce „Tune drive“ je nabídka „Modes of operation“ umožňující volbu jednoho ze čtyř režimů. Cyclic synchronous mode (CSP) uzavírá všechny tři regulační smyčky v měniči EL7221. Cyclic synchronous velocity mode (CSV) uzavírá v měniči rychlostní a proudovou zpětnou vazbu, polohová zpětná vazba je uzavírána skrze NC úlohu. Režimy cyclic synchronous torque mode (CST) a cyclic synchronous torque mode with commutation angle (CSTA) uzavírají v měniči pouze proudovou zpětnou vazbu:

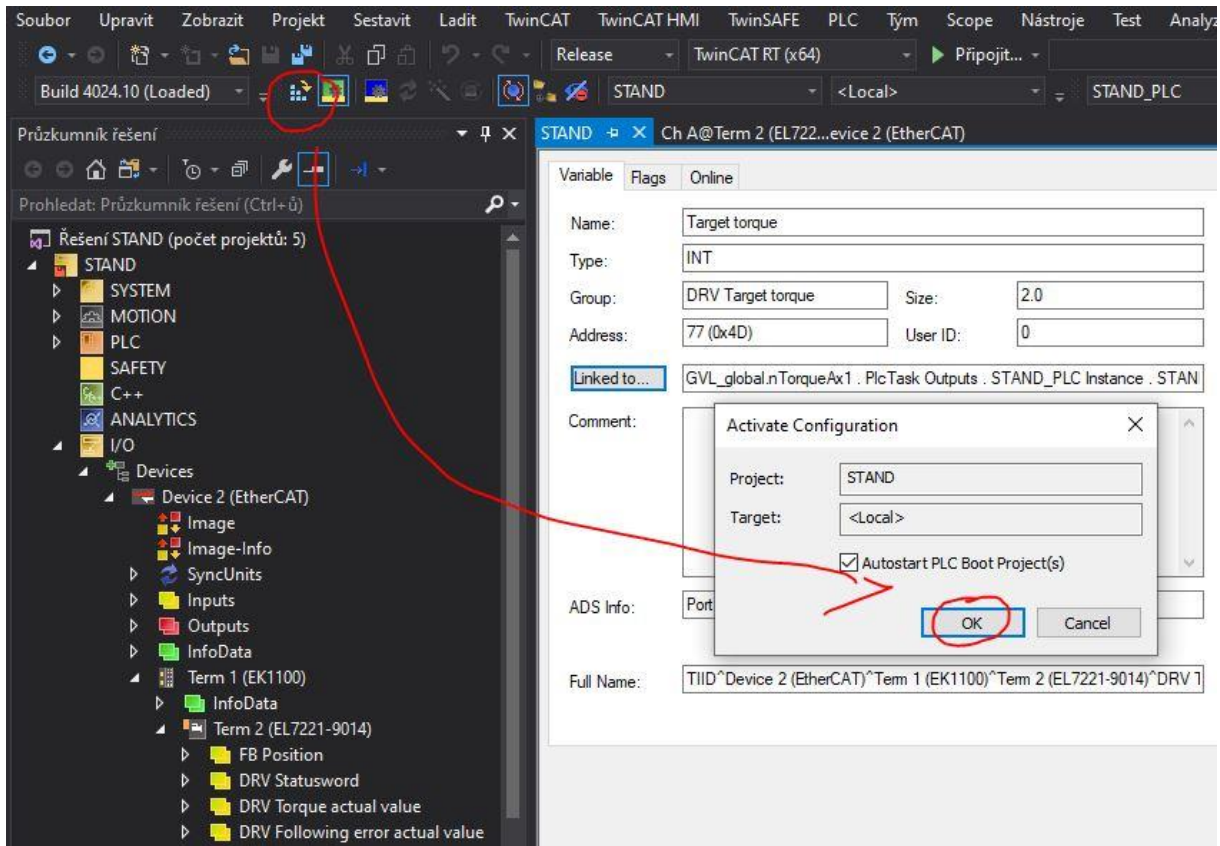


Při používání režimu CST je vždy nutné manuálně namapovat požadovaný moment z PLC na vstup do měniče:



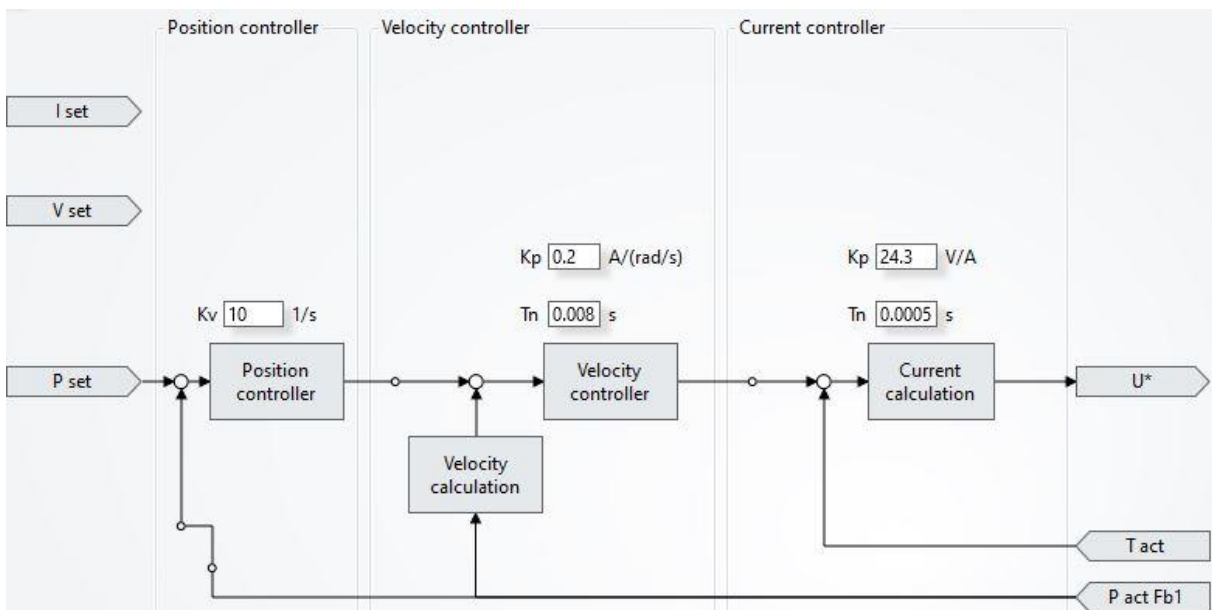
Při každé změně režimu se mapování resetuje, proto je nutné vždy po přepnutí do režimu CST manuálně namapovat požadavek na krouticí moment.

Změna konfigurace se poté aktivuje tlačítkem „Activate configuration“ a potvrzením OK:



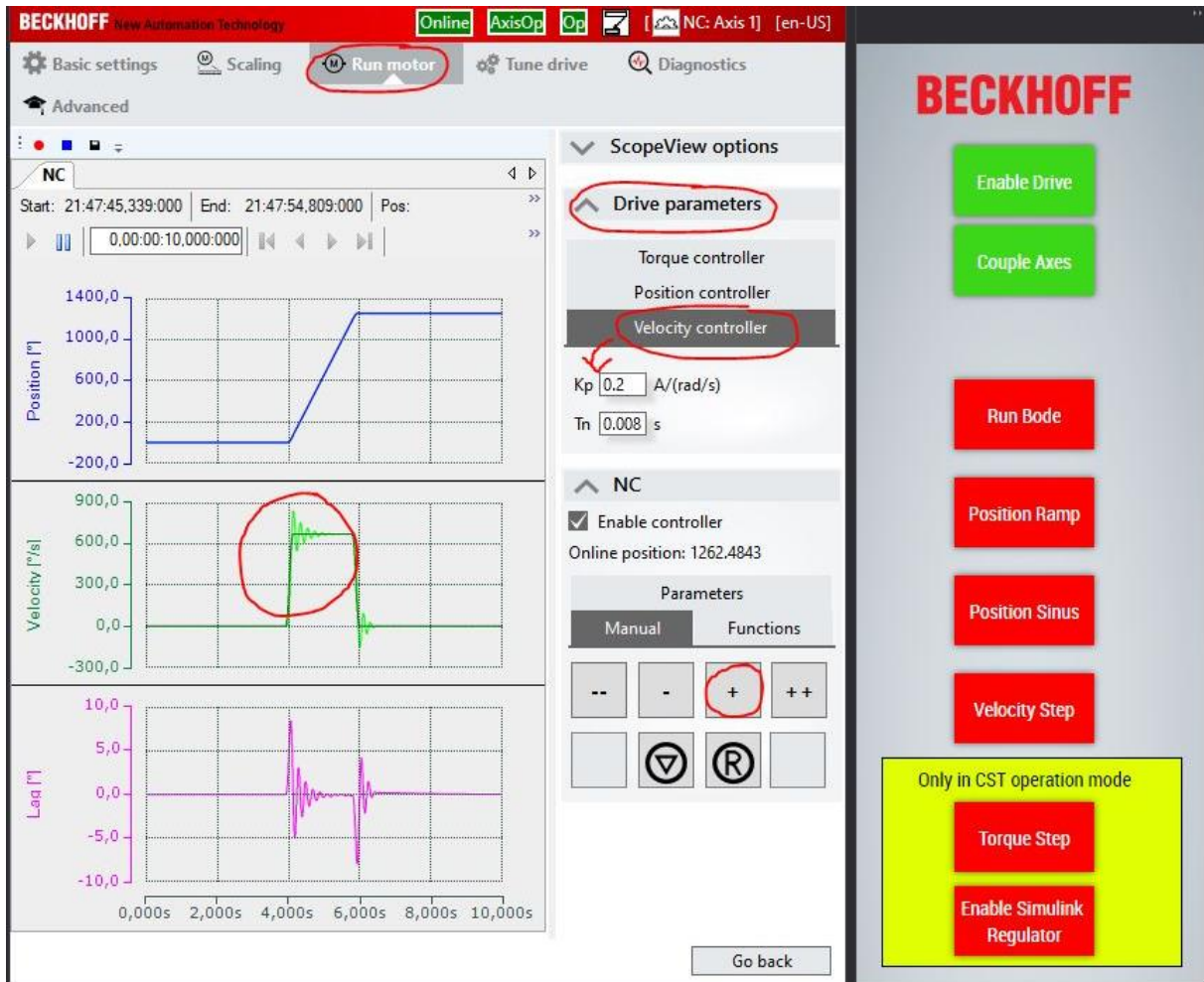
### 3 Ladění regulace

Při každé změně režimu je potřeba opětovně aktivovat konfiguraci. Záložka „Tune drive“ v Drive Manager 2 nabízí pohled na celou kaskádní regulaci včetně aktuálně nastavených regulačních konstant:

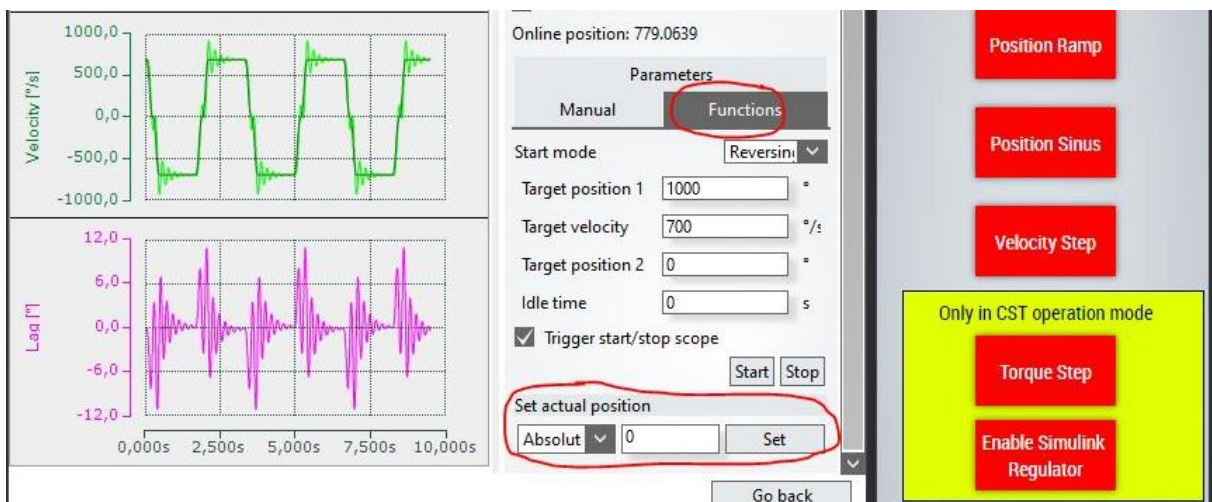




Ladění regulace probíhá v záložce „Run motor“. Pod volbou „Drive parameters“ je možné zvolit konstanty polohového, rychlostního a proudového regulátoru a libovolně je měnit. Zvolená hodnota se potvrdí a nahraje do měniče stisknutím Ctrl + Enter. Pro pohyby lze využívat tlačítka „+“ a „-“ ve volbě „Manual“:



Pro opakovaný pohyb lze využít volbu „Functions“ ve které lze rovněž vynulovat aktuální polohu pomocí „Set Actual position“:



Při ladění regulace první osy je vhodné provozovat osu v příslušném režimu (CSP, CSV) a druhou osu přepnout do režimu CST. Osa v režimu CST s namapovanou PLC proměnnou má po enablevání na výstupu nulový kroutící moment a odbrzděnou brzdu. Naladěné regulační konstanty je poté možné aplikovat i na druhou osu.

## 4 Přednastavené pohyby os

Pro pohyby os lze použít naprogramované funkční bloky – rampu a sinus polohy, skok rychlosti a skok momentu. Parametry pohybu lze nastavit ve funkčním bloku FB\_MOVEMENT:

The screenshot shows the 'Průzkumník řešení' (Solution Explorer) on the left and the 'FB\_MOVEMENT' parameter list on the right. The parameter list includes:

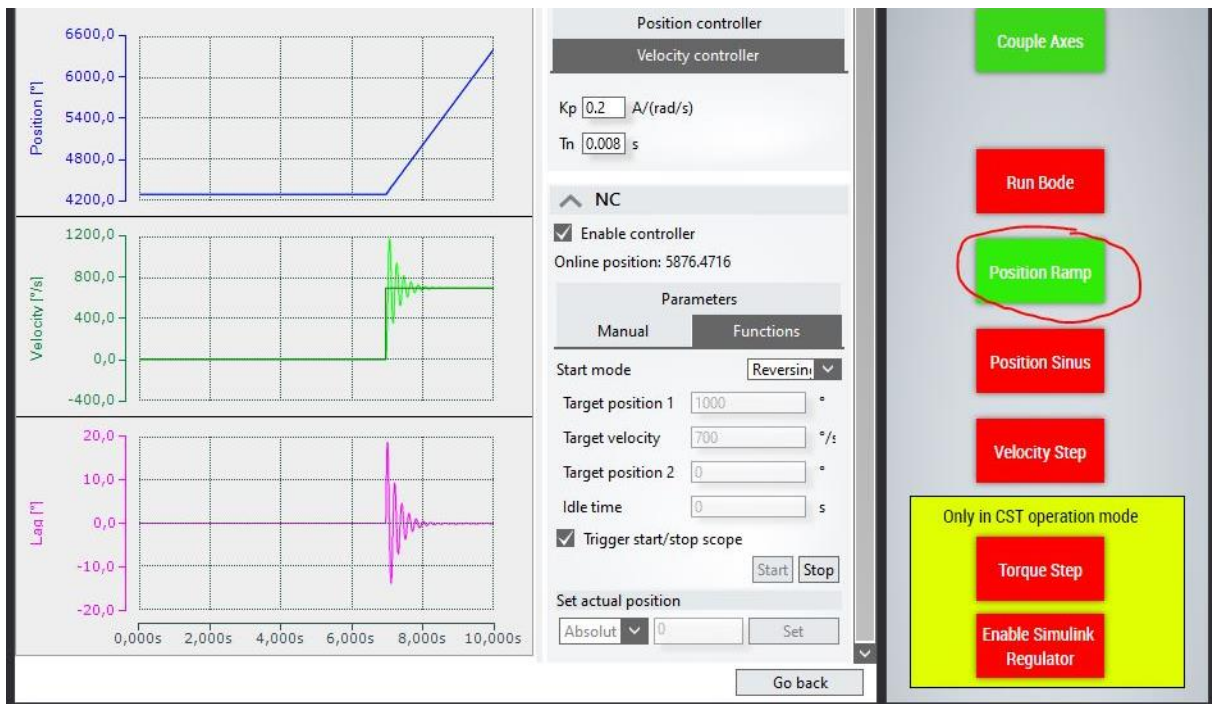
- fPosRampVelo : FLOAT; // Ramp velocity
- fPosRampSlope : FLOAT := 700; // Slope of the ramp
- fPosRampTarget : FLOAT := 100000; // Target position
- fPosSignal : FLOAT; // Position signal
- fSinus : FLOAT; // Sine position
- fSinAmp : FLOAT := 100; // Sine position amplitude
- fSinOffset : FLOAT := 0; // Sine position offset
- tSinPeriod : TIME := T#1S; // Sine position period
- fStep : FLOAT; // Velocity step
- fStepVelo : FLOAT; // Velocity step velocity
- fStepSlope : FLOAT := 4000; // Velocity step slope
- fStepTarget : FLOAT := 100000; // Velocity step target
- fVeloSignal : FLOAT; // Velocity signal
- nSignalID : INT := 1; // ID of the position
- nState : INT := 0; // State of CASE
- stParamsDerAcc : ST\_CTRL\_DIGITAL\_FILTER\_PARAMS; // Parameter
- stParamsDerVelo : ST\_CTRL\_DIGITAL\_FILTER\_PARAMS; // Parameter
- stParamsRamp : ST\_CTRL\_RAMP\_GENERATOR\_EXT\_PARAMS; // Parameter
- stParamsSinus : ST\_CTRL\_SIGNAL\_GENERATOR\_PARAMS; // Parameter
- stParamsStep : ST\_CTRL\_RAMP\_GENERATOR\_EXT\_PARAMS; // Parameter
- tRampTime : TIME := T#5S; // Ramp driving time
- tSinusTime : TIME := T#5S; // Sine driving time
- tStepTime : TIME := T#5S; // Step driving time

Velikost skoku kroutícího momentu je nastavitelná ve funkčním bloku FB\_STEP\_TORQUE:

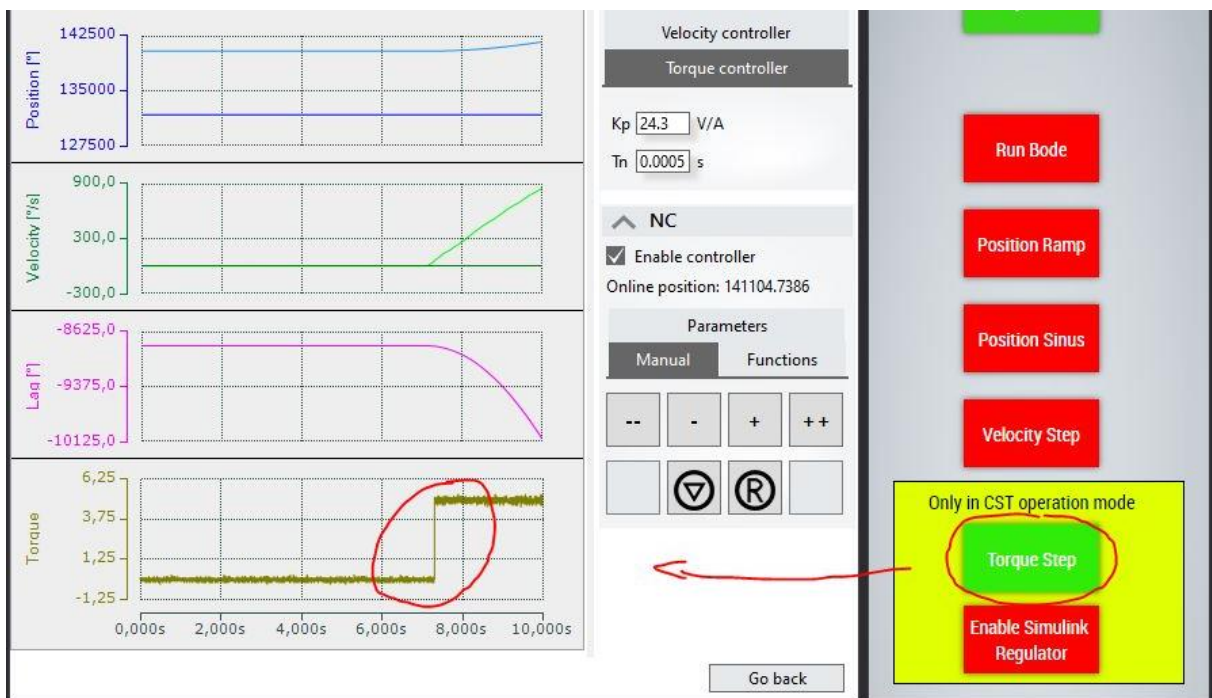
The screenshot shows the 'Průzkumník řešení' (Solution Explorer) on the left and the 'FB\_STEP\_TORQUE' parameter list on the right. The parameter list includes:

- bDone : BOOL := FALSE;
- VAR\_IN\_OUT
- Axis : AXIS\_REF;
- VAR
- // Variables
- bWait : BOOL;
- bMove : BOOL;
- nState : INT := 0;
- nTorqueSetPoint : INT := 50;
- tWait : TIME := T#1S;
- tRunStepTorque : TIME := T#4S;
- // Function blocks
- rTRIGstep : R\_TRIG;
- TONtimerWait : TON;
- TONtimerMove : TON;

Pohyb je spuštěn kliknutím na příslušné pole v HMI. Pokud je pohyb vykonáván, změní se barva tlačítka na zelenou. Přednastavená tlačítka mohou urychlit ladění regulace rozšířením rychlé volby pohybů z Drive Manager 2:



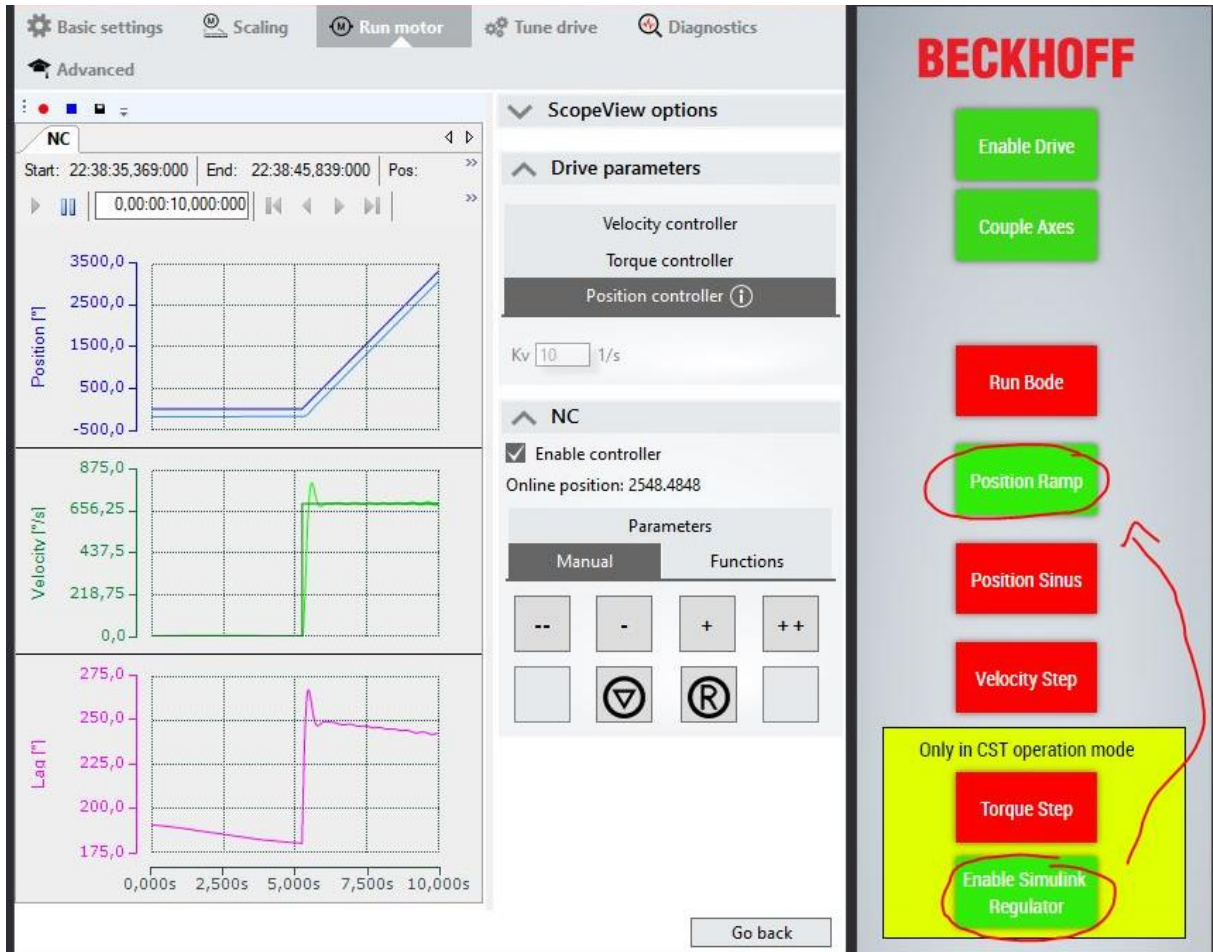
Pro skok momentu „Torque Step“ je nutné přepnout obě osy do režimu CST, na první osu namapovat PLC proměnnou GVL\_global.nTorqueAx1 a aktivovat konfiguraci. Skok momentu se spustí tlačítkem „Torque step“:



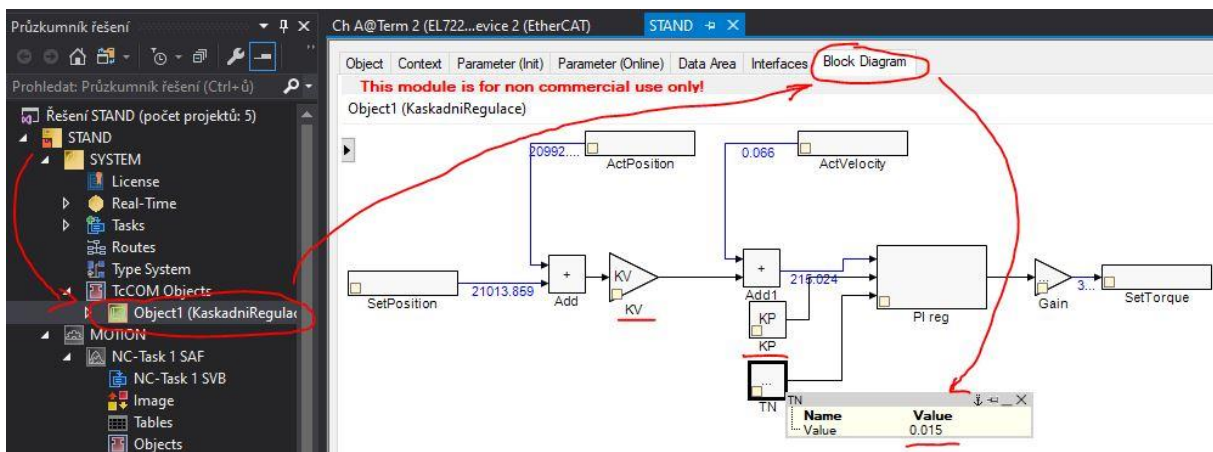


## 5 Ladění regulátoru ze Simulinku

Pro spuštění regulátoru vytvořeného v Simulinku je nutné přepnout pohony do režimu CST a namapovat PLC proměnnou GVL\_global.nTorqueAx1. Následně je možné tlačítkem „Enable Simulink Regulator“ spustit regulátor. Po spuštění regulátoru je možné zadávat požadavky na pohyb běžným způsobem:



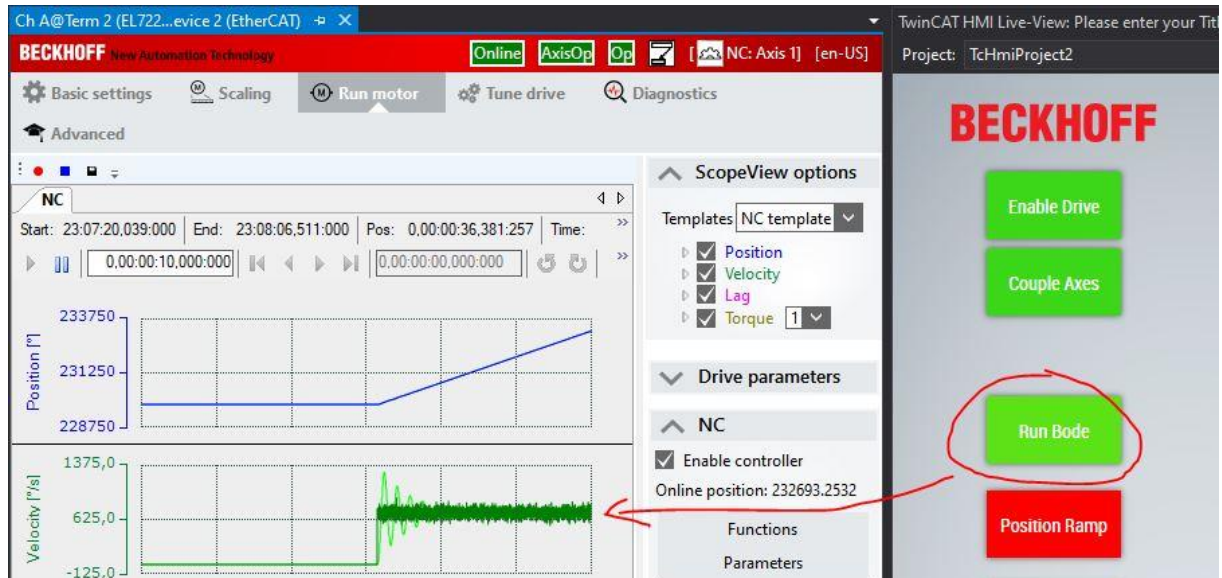
Regulátor je možné ladit v TcCOM Objects/Objekt1 v záložce „Block Diagram“:



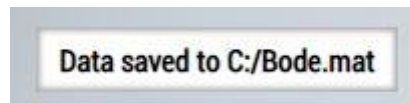
Volbu lze opět potvrdit kombinací kláves Ctrl + Enter.

## 6 Měření Bode

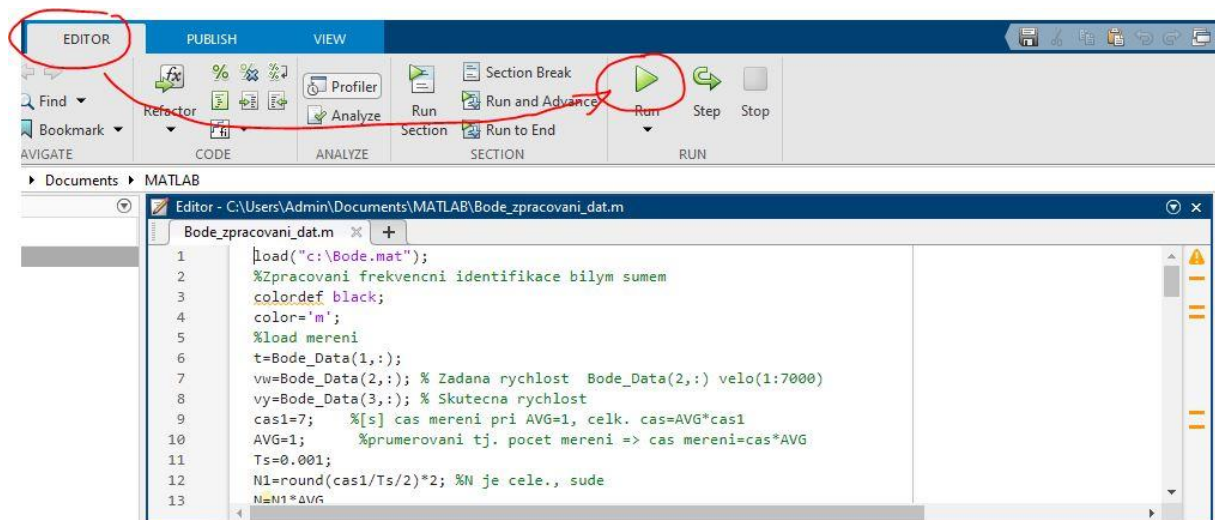
Pro měření přenosu rychlostní regulace první osy by měla být osa přepnutá do režimu CSV. Druhá osa je v režimu CST. Spuštění měření se provede tlačítkem „Run Bode“:



Po naměření se zobrazí místo, kam se data uložila:



Vyhodnocení je možné spuštěním Matlab skriptu, který je uložený na ploše pod názvem „Bode\_zpracovani\_dat.m“ a který automaticky načte naměřená data z TwinCATu:



Po provedení výpočtu se zobrazí Bode diagram přenosu:

