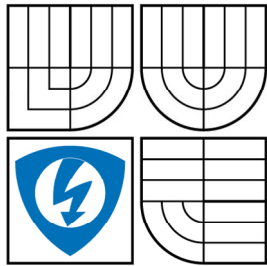
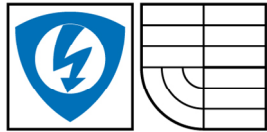


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV AUTOMATIZACE A MERICÍ TECHNIKY



FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

REALIZACE APLIKACE PRO OVLÁDÁNÍ SERVOPOHONU.

REALIZATION OF THE APPLICATION FOR CONTROL OF THE SERVO DRIVES.

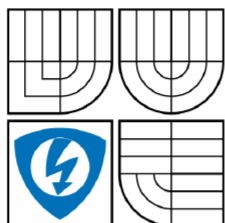
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

ZBYNĚK POSPÍCHAL

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR MALOUNEK



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav automatizace a měřicí
techniky

Bakalářská práce

bakalářský studijní obor
Automatizační a měřicí technika

Student: Pospíchal Zbyněk

ID: 74910

Ročník: 3

Akademický rok: 2007/2008

NÁZEV TÉMATU:

Realizace aplikace pro ovládání servopohonu.

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s možnostmi a použitím servozesilovačů ACOPOS firmy B&R. Prostudujte možnosti řízení pomocí knihovny PLC-Open. Dále se seznamte s vizualizačními prostředky B&R. Pomocí PLC s vestavěným dotykovým panelem navrhnete aplikaci, jejíž pomocí budete schopni ovládat synchronní motor.

Prostudujte a ověřte možnosti řízení motoru přímo pomocí vlastního algoritmu.

DOPORUČENÁ LITERATURA:

Termín zadání: 1.2.2008

Termín odevzdání: 2.6.2008

Vedoucí práce: Ing. Petr Malounek

prof. Ing. Pavel Jura, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

LICENČNÍ SMLOUVA

POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: Zbyněk Pospíchal
Bytem: Vítězslavy Kaprálové 899/26, 61800, Brno -
Černovice
Narozen/a (datum a místo): 12. 6. 1986, Brno

(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií
se sídlem Údolní 244/53, 602 00, Brno
jejímž jménem jedná na základě písemného pověření děkanem fakulty:
doc. Ing. Václav Jirsík, CSc.
(dále jen „nabyvatel“)

Čl. 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):
 - disertační práce
 - diplomová práce
 - bakalářská práce
 - jiná práce, jejíž druh je specifikován jako(dále jen VŠKP nebo dílo)

Název VŠKP: Realizace aplikace pro ovládání servopohonu.
.....
Vedoucí/ školitel VŠKP: Ing. Petr Malounek
.....
Ústav: Ústav automatizace a měřicí techniky
.....
Datum obhajoby VŠKP:

VŠKP odevzdal autor nabyvateli v*:

- tištěné formě – počet exemplářů 1
- elektronické formě – počet exemplářů 1

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

* hodící se zaškrtněte

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne:

.....
Nabyvatel

.....
Autor

Anotace

Cílem této bakalářské práce je představit servozsilovač ACOPOS firmy B&R. Jsou vysvětleny jednotlivé části a softwarová výbava. Na demonstrační úloze je představeno řízení pomocí knihovny PLCopen. Výsledná aplikace je řízena pomocí PLC s dotykovým panelem s vizualizací procesu. V systému je implementován vlastní regulační algoritmus, který nahrazuje regulátor v servozsilovači.

Klíčová slova

servopohon, ACOPOS, servozsilovač

Annotatio

The goal of this bachelor work is introduce servodrive ACOPOS from B&R company. In the introduction are presented basic components and software equipment. The servodrive is driven by PLCopen library function. For operating is used PLC with Touch Panel with visualization of process. At the end of work was implemented own control algorithm which replace controler in servodrive.

Klíčová slova - anglicky:

servo drive, ACOPOS

Bibliografická citace

POSPÍCHAL, Zbyněk. *Realizace aplikace pro ovládání servopohonu*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 50 s., příloh. Vedoucí práce Ing. Petr Malounek.

P r o h l á š e n í

„Prohlašuji, že svou bakalářskou práci na téma "Realizace aplikace pro ovládání servopohonu" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.“

V Brně dne :

Podpis:

P o d ě k o v á n í

Děkuji tímto svému vedoucímu Ing. Petru Malounkovi za cenné připomínky a rady při vypracování bakalářské práce.

V Brně dne :

Podpis:

OBSAH

1. ÚVOD	- 1 -
2. HARDWARE	- 3 -
2.1 Servozesilovač Acopos	- 3 -
2.2 motor	- 4 -
2.3 encoder	- 5 -
2.4 plc	- 6 -
2.5 komunikace	- 6 -
3. SOFTWARE	- 7 -
3.1 automation studio	- 7 -
3.1.1 Watch	- 8 -
3.1.2 Tracer	- 8 -
4. NC KONCEPT	- 10 -
4.1.1 Servozesilovač	- 10 -
4.1.2 Řízení a komunikace	- 10 -
4.1.3 Software koncept	- 11 -
4.2 Konfigurace	- 12 -
4.2.1 INIT Parametrický modul	- 12 -
4.2.2 ACOPOS Parametrická Tabulka	- 12 -
4.2.3 Deployment Tabulka	- 12 -
5. PLCOPEN	- 14 -
5.1 programování	- 15 -
5.1.1 „naccess“ funkce	- 15 -
5.1.2 Funkční bloky	- 16 -
5.1.3 „naction“ funkce	- 16 -
5.2 Struktura programu	- 17 -
5.2.1 Motion control sample project	- 18 -
5.2.2 Použité funkce	- 19 -
5.2.3 Speciální funkce	- 19 -
6. VIZUALIZACE	- 21 -
6.1 zobrazovací zařízení	- 22 -
6.2 Program pro vizualizaci	- 23 -
7. ACOPOS CONTROL CONCEPT	- 25 -
7.1 Koncepce kaskádního řízení	- 25 -

7.2 generátor žádané hodnoty	- 26 -
7.3 polohový regulátor	- 27 -
7.4 rychlostní regulátor	- 28 -
7.5 proudový regulátor	- 29 -
8. NASTAVENÍ REGULÁTORŮ	- 31 -
8.1 použití vnitřních regulátorů	- 32 -
8.1.1 Nastavení rychlostního regulátoru	- 34 -
8.1.2 Nastavení polohového regulátoru	- 34 -
8.2 implementace vlastního polohového regulátoru	- 36 -
8.2.1 PS regulátor s beznárazovým přepínáním a dynamickým omezením sumační složky	- 37 -
8.2.2 Nastavení parametrů regulátoru	- 38 -
8.3 Srovnání odezvy regulátorů	- 39 -
9. ZÁVĚR	- 42 -
10. LITERATURA.....	- 44 -
SEZNAM ZKRATEK A SYMBOLŮ.....	- 45 -
SEZNAM PŘÍLOH	- 46 -

SEZNAM OBRÁZKŮ

Obrázek 2.1 Smyčka proudového regulátoru [8]	- 5 -
Obrázek 2.2 Resolver interface [9]	- 6 -
Obrázek 3.1 Automation studio	- 7 -
Obrázek 3.2 Prostředí funkce Tracer.....	- 9 -
Obrázek 4.1 Komunikace NC manager a ACOPOS [3]	- 11 -
Obrázek 4.2 Deployment Tabulka	- 13 -
Obrázek 5.1 Funkční blok [3]	- 16 -
Obrázek 5.2 Stavový diagram úlohy [3]	- 18 -
Obrázek 6.1 Prostředí pro tvorbu vizualizační aplikace	- 22 -
Obrázek 6.2 Power Panel [12].....	- 23 -
Obrázek 6.3 Stránka pro zadání parametrů PS regulátoru, stránka pro nastavení	- 24 -
Obrázek 6.4 Stránka pro ovládání pohonu	- 24 -
Obrázek 7.1 Celková regulační smyčka [8]	- 25 -
Obrázek 7.2 Kaskádní řazení regulátorů [5]	- 26 -
Obrázek 7.3 Polohový regulátor [5]	- 27 -
Obrázek 7.4 Rychlostní regulátor [5]	- 28 -
Obrázek 8.1 Odezva vnitřních regulátorů $v_{\max} = 1000$ jedn./s.....	- 35 -
Obrázek 8.2 Odezva vnitřních regulátorů $v_{\max} = 3000$ jedn./s.....	- 35 -
Obrázek 8.3 Struktura PS regulátoru.....	- 37 -
Obrázek 8.4 Odezva PS regulátoru polohy a vnitřního rychlostního reg.....	- 39 -
Obrázek 8.5 Srovnání průběhů obou regulátorů	- 40 -

SEZNAM TABULEK

Tabulka 5.1 Parametry funkce „ncacces“ [4].....	- 15 -
Tabulka 5.2 Parametry funkce „ncaction“ [4].....	- 17 -
Tabulka 8.1 Použité vnitřní parametry [7]	- 32 -
Tabulka 8.2 Parametry motoru a servozsilovače [9]	- 33 -
Tabulka 8.3 Parametry rychlostního regulátoru	- 34 -
Tabulka 8.4 Parametry polohového regulátoru	- 34 -
Tabulka 8.5 Parametry regulátoru podle metody Ziegler-Nichols.....	- 38 -
Tabulka 8.6 Parametry PS regulátoru	- 39 -

1. ÚVOD

V poslední době dochází k rozmachu elektrických servopohonů v automatizaci. Díky rozvoji ve výkonové a výpočetní technice zabírá využití servopohonů se střídavými motory dominantní postavení na trhu, donedávna téměř výhradně využívané stejnosměrné motory jsou postupně vytlačovány a předpokládá se téměř vymizení z trhu.

Tato bakalářská práce se zabývá popisem jednotlivých součástí, jež jsou nezbytné pro vytvoření aplikace pro polohové řízení. Praktická část byla realizována na zařízeních od firmy B&R, konkrétně se jedná o synchronní motor, který je napájen ze servozsilovače ACOPOS, jenž v sobě spojuje funkce pro napájení motoru a jeho řízení použitím tří kaskádně zapojených regulátorů. Ovládání motoru je realizováno na PLC Power Panel s dotykovou obrazovkou s vizualizací procesu.

V druhé kapitole je popsána hardwarová část zařízení, výčet možných systémů se omezuje jen na konkrétní zařízení, které jsou použita v demonstrační úloze. Čtenáři je poskytnut popis principů funkce synchronního motoru. Pro realizaci zpětné vazby je ke snímání polohy použit resolver. Dále je pojednáno o výkonové části servozsilovače. Řídící část obsahuje tři regulátory, z nichž proudový regulátor na nejnižší úrovni vytváří řídicí impulsy pro střídač, jenž napájí motor pulsně šířkově modulovaným signálem.

Následující kapitola popisuje prostředí automation studio pro vývoj aplikace a také komponenty umožňující snadnou tvorbu vizualizace, která je použita při ovládání motoru.

Stěžejní část práce se zabývá popisem a realizací aplikace pro řízení servopohonu pomocí knihovny PLCopen. Před samotnou realizací jsou popsány součásti prostředí NC koncept, zahrnující veškeré potřebné součásti pro řízení servopohonu. Je pojednáno o NC manager, což je operační systém pro PLC vytvářející spojení mezi uživatelskou aplikací a operačním systémem, který řídí servozsilovač. Pro konfiguraci systému je využita deployment tabulka a moduly inicializačních parametrů. Samotný řídicí program využívá standardní funkce PLCopen. Program obsahuje příkazy pro nastavení parametrů pohybu a typ regulace na cílovou vzdálenost nebo rychlost, dále obsahuje i zobrazení chybových stavů.

Při návrhu systému řízení představuje kritickou úlohu správné nastavení parametrů regulátorů, v kapitole šest jsou popsány vnitřní regulátory implementované v servozsilovači, konkrétně se jedná o PI regulátory, diskrétně realizované, s určitými modifikacemi podle použití jako regulátor polohy, rychlosti, proudu. Následující kapitola pojednává o konkrétním nastavení regulátorů s využitím empirických vztahů a dále s optimalizací nastavení s ohledem na tvar odezvy při změně žádané hodnoty.

V poslední části je nastíněna možnost náhrady vnitřního polohového regulátoru v servozesilovači vlastním regulačním algoritmem běžícím na řídicím PLC. Pro potřebnou změnu ve struktuře regulátorů je zapotřebí použití funkcí specifikovaných firmou B&R, tyto funkce mají stejnou strukturu, ale nejsou zahrnuty ve standartu PLCopen. Nahrazení regulátoru je umožněno použitím realtime sběrnice ETHERNET Powerlink. K porovnání chování byl zvolen PS regulátor s dynamickým omezením sumační složky. V poslední části jsou porovnány chování obou typů regulátorů z hlediska průběhů akčních zásahů při přechodném ději.

2. HARDWARE

Aplikace pro ovládání servopohonu obsahuje, z pohledu hardwarové konfigurace tyto základní prvky: servozsilovač jako ústřední člen spojující v sobě funkci napájení a řízení samotného motoru a funkci pro ovládání tohoto motoru z nadřazeného systému, v tomto případě PLC. Pro realizaci zpětné vazby lze použít řadu snímačů implementovaných přímo na osu motoru nebo na technologii. Pro komunikaci mezi servozsilovačem a nadřazeným systémem lze použít různé typy sítí. Tato kapitola si klade za cíl výčet konkrétních prvků použitých v demonstrační úloze.

2.1 SERVOZESILOVAČ ACOPOS

Srdcem systému je servozsilovač, konkrétně ACOPOS 1022. Tento sofistikovaný systém pro řízení servopohonu obsahuje prostředky pro napájení motoru a pro jeho řízení, za použití tří kaskádně zapojených regulačních smyček. Pro řízení proudu motorem jsou na nejnižší úrovni implementovány tři regulátory proudu pro každou napájenou fázi. Nad tímto regulátorem funguje rychlostní a dále pak polohový regulátor. Přesný popis struktury a nastavení příslušných regulátorů bude probrán v další kapitole. Primární funkcí servozsilovače je převod síťového napájecího napětí odebíraného z třífázové sítě a jeho transformace na napětí vhodné k napájení motoru. Pro napájení synchronního motoru se využívá pulsně šířkové modulace, což přináší vysokou flexibilitu pro generování napěťové charakteristiky.

Převod napětí se děje ve třech částech. Odebírané harmonické napětí ze sítě je usměrněno v usměrňovači a přivedeno do stejnosměrné části, která obsahuje kondenzátor jako zdroj napětí, součástí je i brzdový rezistor, na kterém je spotřebována energie při brzdění motoru. Následuje střídač v šestipulsním zapojení, tvořený plně říditelnými výkonovými tranzistory IGBT. Řízení jejich spínání obstarává proudový regulátor podle polohy rotoru a žádané hodnoty nastavené jako výstup z rychlostního regulátoru. Toto uspořádání umožňuje pracovat ve čtyřkvadrantovém režimu, což je vlastnost nezbytná pro použití v servořízení.

K zabránění tepelného zničení výkonových tranzistorů střídače a vinutí motoru jsou na exponovaných místech implementovány snímače teploty, na tranzistorech není možno měřit teplotu přímo na součástkách, z tohoto důvodu je v servozsilovači počítán matematický model, který vystihuje oteplování jednotlivých komponent.

Servozsilovač je vybaven vlastním operačním systémem, který se stará o řízení, nastavování jednotlivých parametrů a komunikaci s nadřazeným řídicím systémem.

Součástí je i několik digitálních vstupů a dva sloty pro výsuvné karty, které obsahují interface pro komunikaci, připojení enkodérů nebo modul s CPU.

2.2 MOTOR

Zastává roli akčního členu, jeho vlastnosti zásadním způsobem ovlivňují výsledné chování celé mechatronické soustavy. S rozvojem polovodičových prvků vhodných pro aplikace výkonové elektroniky, hlavně rozvoj plně říditelných součástek, tranzistorů IGBT, je možné nahradit stejnosměrné motory, které díky svým vlastnostem vyhovovaly požadavkům servosystémů. Tyto požadavky jsou vynikající dynamické vlastnosti a snadné řízení. Nevýhodou je jejich konstrukce, kdy dochází k mechanickému opotřebení kartáčů nutných pro napájení rotoru. Tyto nedostatky jsou odstraněny použitím motorů střídavých. Dnes jsou v polohových aplikacích hojně využívány synchronní motory.

Synchronní motor je točivý stroj, který mění elektrickou energii v energii mechanickou. Stator je opatřen napájecím vynutím, které je napájeno z měniče s proměnným kmitočtem. Rotor je v případě servoaplikací nejčastěji opatřen pólovými nástavci s permanentními magnety. Napájením statorového vinutí sinusovým proudem vzniká v mezeře mezi statorem a rotorem točivé magnetické pole. Pro dosažení maximálního krouticího momentu a tím i nejvyšší účinnosti je nezbytné, aby se rotor točil synchronně s tímto magnetickým polem. Při řízení motoru je tedy nezbytné mít trvalou informaci o poloze rotoru.

Elektromagnetický moment synchronního motoru je dán vztahem

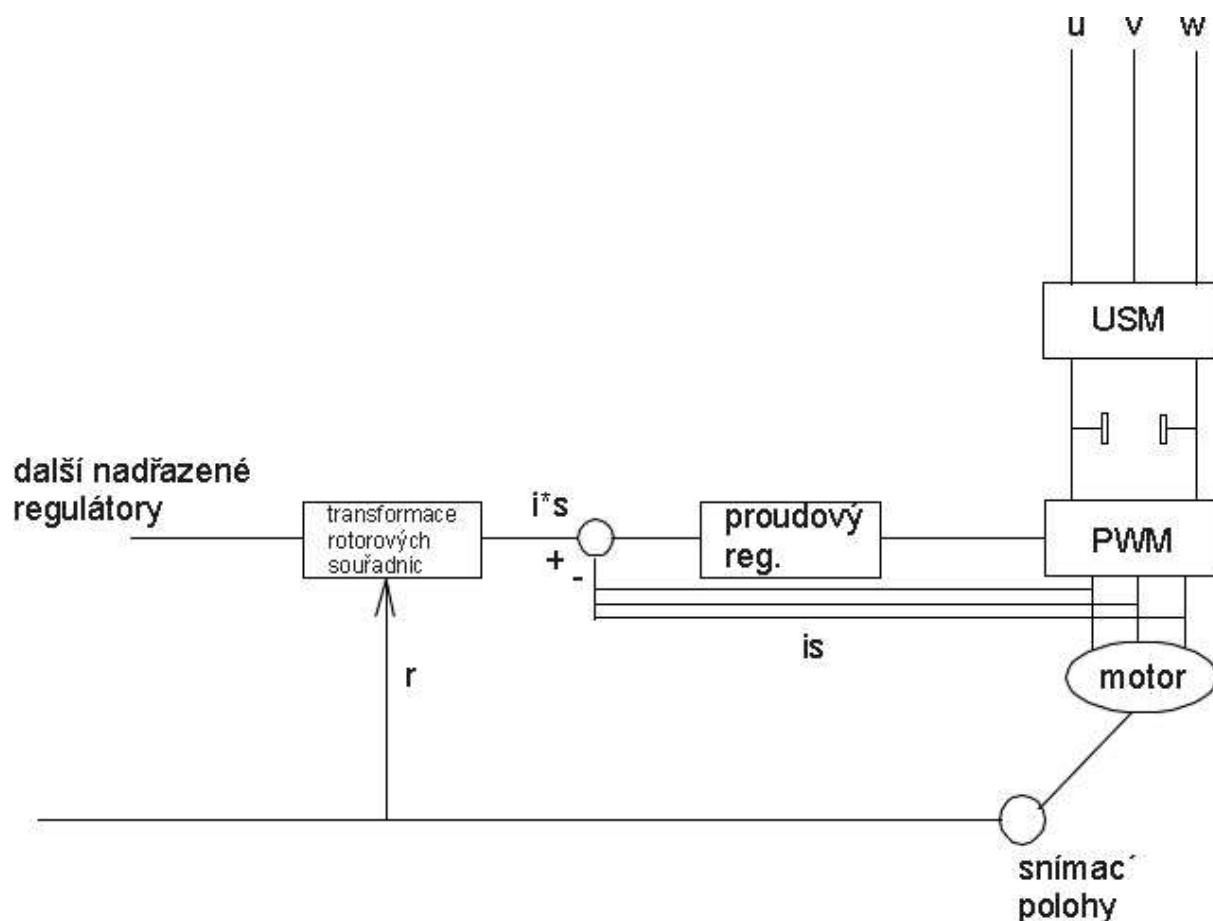
$$M = p \cdot \Psi_m |i_s| \cdot \sin \delta \quad (2.1)$$

δ – úhel vektoru statorového proudu a magnetického toku, pro permanentní magnety je magnetický tok totožný s polohou rotoru.

p – počet pólových dvojic synchronního motoru

Ψ_m – spřažený magnetický tok permanentních magnetů

Pro motory buzené permanentními magnety na rotoru je vektor magnetického toku totožný s polohou rotoru.



Obrázek 2.1 Smyčka proudového regulátoru [8]

V případě této aplikace je použit synchronní motor od firmy B&R, jedná se o třífázový, šestipólový motor napájený ze servozesilovače. Napájecí napětí má sinusový průběh ovládaný pulsně šířkovou modulací. Při napájení ze sítě 3x480V/50Hz lze dosáhnout až 3000 otáček za minutu a maximální krouticí moment 115 Nm. Statorové vyunutí je opatřeno snímačem teploty pro eliminaci tepelné destrukce vlivem zahřátí procházejícím proudem při přetížení. Motor je navržen s ohledem na minimalizaci setrvačných ztrát, tím se dosahuje zlepšení dynamických vlastností.

2.3 ENCODER

Pro realizaci zpětné vazby při řízení pohybu a pro samotnou funkci synchronního motoru je nezbytné mít informaci o poloze rotoru. Za tímto účelem se používají měřicí elementy, které mohou být indukční, optické a jiné. Rozlišení těchto zařízení je jedním z nejdůležitějších kritérií, které ovlivňují kvalitu regulace a přesnost výsledného servosystému.

Servozesilovače ACOPOS umožňují použitím výsuvných karet připojení různých typů čidel polohy.

V případě této úlohy je použit resolver. Je to indukční absolutní snímač, jeho předností je robustní a jednoduchá konstrukce. Funkčně se jedná o rotační transformátor. Rotorové vinutí je napájeno sinusovým signálem o konstantní frekvenci. Stator tvoří dvojice vinutí navzájem posunutá o 90° , do kterých se indukují napětí ze statoru. Při pohybu se na výstupu statorových vinutí objeví sinusový signál, jehož průběh je na výstupu z obou vinutí fázově posunut o 90° , procesní logika tento posun vyhodnotí a určí polohu. Snímač je umístěn na hřídeli motoru a procesní logika s A/D převodníkem je umístěna na výsuvné kartě, která se zasouvá do slotu v servozesilovači.



Obrázek 2.2 Resolver interface [9]

2.4 PLC

Pro ovládání servozesilovače bylo zvoleno PLC s dotykovou obrazovkou od firmy B&R, Power Panel PP282. Na tomto zařízení běží vizualizační aplikace, která slouží pro ovládání motoru. Spojení s počítačem je použito klasického ethernetu TCP/IP.

2.5 KOMUNIKACE

Podle použitého výsuvného modulu je možné používat ke komunikaci s nadřazeným systémem, nebo pro připojení do průmyslové sítě standart CAN nebo firemní úpravu společnosti B&R protokolu ethernet, ETHERNET Powerlink. V aplikaci je použit ETHERNET Powerlink. Jeho specifikace umožňuje použití v realtime aplikacích a jeho rychlost ho předurčuje k použití v časově kritických úlohách jakou je i servořízení.

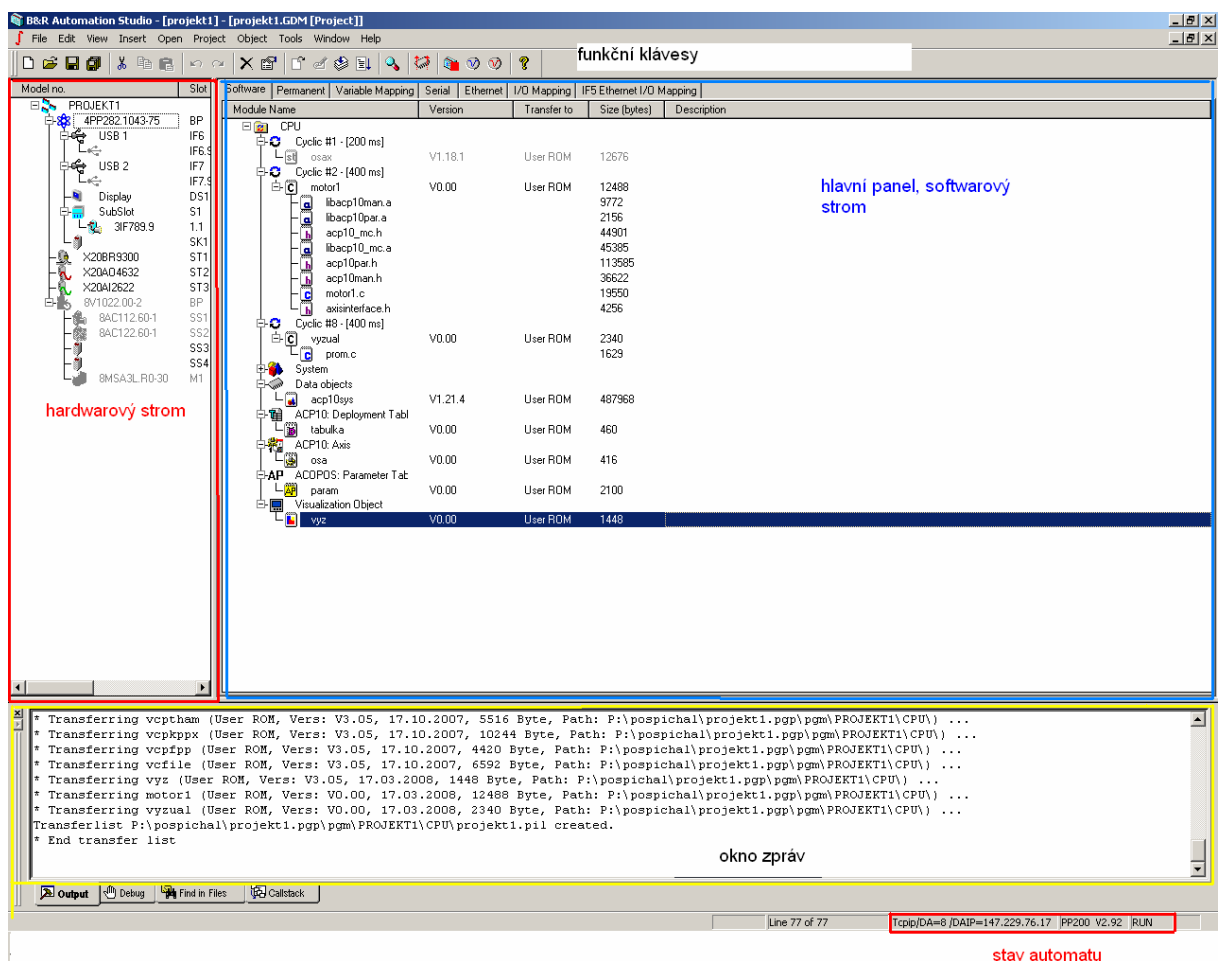
3. SOFTWARE

3.1 AUTOMATION STUDIO

Automation studio je vývojové prostředí pro práci s automaty od firmy B&R. Obsahuje řadu komponent pro tvorbu aplikací programových i vizualizačních. Dále umožňuje sledování aplikace za běhu, zobrazení grafických průběhů jednotlivých proměnných, nástroje pro testování a správu alarmů.

Aplikace je možno psát v těchto programovacích jazycích:

- Ladder diagram (LAD)
- Sequention Function Chart (SFC)
- Instruction list (IL)
- Structured text (ST)
- Automation basic (AB)
- ANSI C (C)



Obrázek 3.1 Automation studio

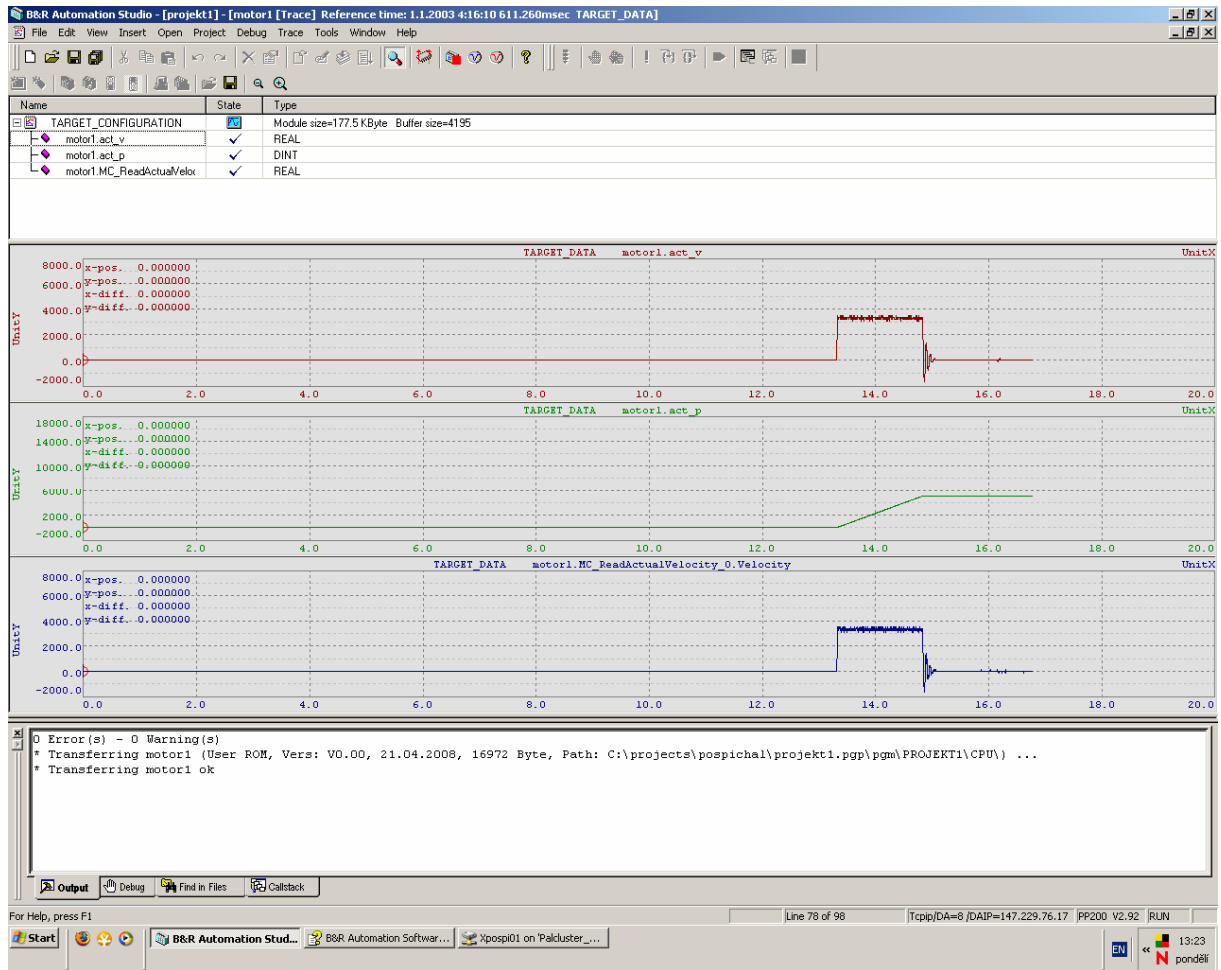
Při zakládání projektu je možno použít automatické rozpoznání připojeného hardwaru. Po úspěšném rozpoznání, nebo ruční konfiguraci se tento objeví v levém okně aplikace, v podobě hardwarového stromu. Hlavní část představuje softwarový strom, do kterého je možné přidávat aplikační programy s přiřazením do jednotlivých cyklických bloků s různým cyklickým časem pro vykonání a s různou prioritou. Další součástí je operační systém a jiné volitelné bloky specifické pro každou aplikaci lišící se použitým hardwarem. Poslední volitelnou součástí je komponenta visual object, která umožňuje jednoduchým způsobem tvorbu vizualizačních aplikací. Po nastavení komunikace ať už pomocí sériové linky nebo ethernetu je na spodní liště zobrazen stav připojeného PLC.

3.1.1 Watch

Watch je nástroj, který umožňuje online zobrazovat a měnit proměnné všech programů běžící na připojeném PLC. Tento nástroj je vhodný pro ladění aplikace.

3.1.2 Tracer

Tracer slouží ke grafickému zobrazení průběhů zvolených proměnných v čase. Komponenta umožňuje manuální spuštění záznamu, nebo spuštění po splnění zadaných podmínek. Naměřené hodnoty je možné dále analyzovat, měnit měřítka zobrazení, vypočítat spektrum signálu atd. Samozřejmostí je uložení naměřených hodnot v několika formátech pro archivaci a další zpracování v běžných kancelářských programech jako MS Excel.



Obrázek 3.2 Prostředí funkce Tracer

4. NC KONCEPT

NC koncept je komplexní systém pro řešení polohových úloh. Systém zahrnuje servozesilovače ACOPOS spojené pomocí sítě ethernet powerlink, nebo CAN, s možností propojení až šestnácti servozesilovačů s jedním řídicím PLC. Systém obsahuje řadu nástrojů pro snadnou konfiguraci komponent a dále software pro implementaci řídicích úloh napsaných podle standardu PLCopen, nebo za využití speciálních funkcí B&R. Pro diagnostiku a testování slouží nástroje jako NC-Test a NC-Tracer.

4.1.1 Servozesilovač

Servozesilovač ACOPOS představuje ve struktuře datový objekt s mnoha parametry. Jednak to jsou parametry nezbytné pro správnou konfiguraci připojeného hardwaru jako charakteristiky motoru, typ použitého snímače polohy a jiné. Tyto parametry jsou použity při inicializaci. Další řada parametrů je nastavována a používána při provádění řídicích sekvencí. Servozesilovače ACOPOS obsahují vnitřní logiku s vlastním operačním systémem *NC operation system*.

Jednotlivé součásti *NC operation systemu* obsahují například:

- Regulátory
- Kontrolní logika
- Generátor žádané hodnoty
- ACOPOS parametr...

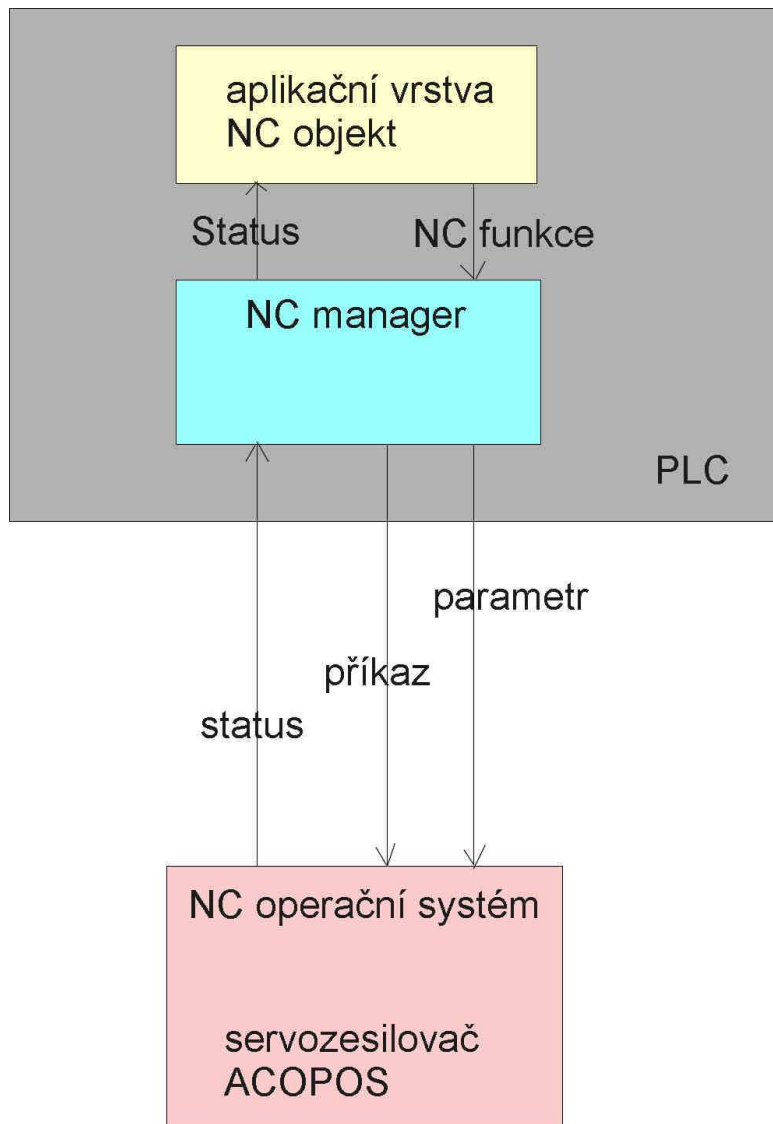
4.1.2 Řízení a komunikace

Na řídicím PLC běží operační systém *NC manager*, který vytváří spojení mezi uživatelským programem a *NC operation system*, operačním systémem implementovaným v servozesilovači. Při inicializaci vytváří *NC manager* datovou strukturu *NC structure* pro každou řízenou osu. Objektově orientovaná struktura *NC structure* člení všechny parametry do logických skupin, vzniká stromová struktura v prostředí objekt → subjekt → akce.

NC manager vykonává tři druhy akcí:

- Koordinace pohybu – (jednoduchý pohyb, CNC, vačkové profily)
- Vyčíslení parametrů a transfer do servozesilovače
- Čtení stavu a zápis do datové struktury

Pro obousměrnou komunikaci lze využít standardních funkcí.



Obrázek 4.1 Komunikace NC manager a ACOPOS [3]

4.1.3 Software koncept

Základem je datový objekt, který popisuje jednotlivou osu. S tímto objektem je pracováno v celé linii, tak jak je nastíněna výše. NC koncept obsahuje tři základní typy datových objektů pro popis os.

- Reálná osa – slouží k popisu vlastností jednotlivé reálné osy reprezentované jedním servozesilovačem a jedním motorem
- Virtuální osa – nemá fyzický ekvivalent a slouží jako generátor hodnot pro reálnou osu, nebo je využívána jako referenční při synchronizaci víceosých systémů

- CNC systém – využívá se pro víceosé aplikace, pro synchronizaci mezi jednotlivými osami

NC koncept plně podporuje využití funkcí standartu PLCopen, pro řízení dalších funkcí a parametrů servozsilovače ACOPOS je přidána řada funkcí vytvořených firmou B&R. Veškeré funkce jsou zahrnuty v knihovně ACP10_MC.

4.2 KONFIGURACE

Systém umožňuje vedle běžné manuální konfigurace všech prvků, kdy je nutné po přidání zařízení do hardwarového stromu ručně zadat jeho parametry, i konfiguraci pomocí softwarových nástrojů, jež umožňují vytvoření předpisu parametrů, které je možné použít pro více zařízení. Toto řešení přináší značnou úsporu času a zpřehledňuje výsledný projekt.

4.2.1 INIT Parametrický modul

Tento softwarový modul je použitelný jak při manuální konfiguraci, tak i při použití flexibilní konfigurace za použití deployment tabulky. Modul definuje základní parametry použité při inicializaci servozsilovače, jako jsou hodnoty nastavení regulátorů, velikosti rychlosti a zrychlení pro základní pohyby. Tento modul lze definovat i pro virtuální osu.

4.2.2 ACOPOS Parametrická Tabulka

Tato komponenta umožňuje přímou adresaci všech parametrů servozsilovače. Každý jednotlivý parametr je pojmenován svým unikátním číslem ID, podle kterého je možné ho identifikovat a měnit jeho hodnotu. Typicky tento modul obsahuje skupiny parametrů jako například „Motor“, „Resolver“ a jiné. Pomocí unikátních ID lze pracovat i s jinak nepřístupnými parametry, jako jsou hodnoty akčních zásahů jednotlivých regulátorů a jiné. Přenastavené hodnoty se projeví po inicializaci softwaru v servozsilovači při použití deployment tabulky nebo vyvoláním specifické funkce za běhu aplikace.

4.2.3 Deployment Tabulka

Umožňuje jednoduše definovat hardware za použití existujících parametrizačních modulů, které mohou být použity pro více os.

Obsahuje tyto položky:

- „NC Object name“ : unikátní symbolické jméno objektu osy
jméno je užíváno v aplikaci při práci s objektem
- „Network interface“ : typ komunikace: Ethernet Powerlink, CAN
- „Node Number“ : číslo modulu systém umožňuje spojit až 16 jednotek ACOPOS s jedním řídicím PLC

- „Advance“ : další volitelné parametry
- „NC Object Type“ : typ objektu osa: reálná, virtuální osa
- „NC INIT Parametr“ : modul inicializačních parametrů
- „ACOPOS Parametr“: definuje parametrickou tabulku,
jejíž data jsou použita při inicializaci
- „Additional Data“ : data pro další funkce
- „Remark“ : poznámky

Toto řešení přináší do projektu větší přehlednost a flexibilitu, s výhodou je v aplikaci při práci s osou použito její symbolické jméno.

NC Object Name	Network Interface	Node Number	Advanced	NC Object Type	Channel	NC INIT Parameter	ACOPOS Parameter	Additional Data
osa1	SL1.S51.IF2 (Powerlink)	1	Default	ncAXIS	1	osa	param	

Obrázek 4.2 Deployment Tabulka

5. PLCOPEN

PLCopen je mezinárodní organizace zabývající se standardizací softwarových prostředků v automatizaci. Cílem je umožnit vývoj softwaru nezávislého na hardwarovém vybavení. Tím to je dosaženo snadné přenositelnosti samotné aplikace a zkracuje se čas potřebný pro vývoj. Standart se zabývá řadou aplikací, mimo jiné obsahuje i část *PLCopen Motion control*, zabývající se řízením polohových úloh.

PLCopen podporuje tyto programovací jazyky:

- Sequential Function Chart (SFC): jazyk použitý pro strukturu vnitřní organizace programů a dále jsou podporovány tyto čtyři nezávislé programovací jazyky:
 - Instruction List (IL)
 - Ladder Diagram (LD)
 - Function Block Diagram (FBD)
 - Structured Text (ST)

Dekompozice a logické členění umožňují modularitu vytvářených programů.

Motion control obsahuje knihovnu funkčních bloků, která definuje základní funkce pro práci se servosystémy, součástí je i definice funkcí pro řízení víceosých systémů a v neposlední řadě standart obsahuje i rozšíření základních funkcí pro proces návratu do výchozí definované pozice, toto rozšíření obsahuje softwarové nástroje a modely pro aplikace.

Pro řízení servozsilovačů ACOPOS je knihovna „*ACP10_MC*“, která obsahuje funkce podle standartu PLCopen Motion control, dále rozšířena o specifické funkce. Pro jejich snadné rozlišení je dodržována konvence jmen funkcí:

- „*MC_*“ - pro funkce standartu PLCopen
 - „*MC_MoveVelocity*“
- „*MC_BR_*“ – pro specifické funkce, které nejsou standardem
 - „*MC_BR_ReadParID*“

Knihovna „*ACP10_MC*“ definuje funkce pro jednoosé systémy jako je nastavení a vykonání pohybů na cílovou vzdálenost, na požadovanou rychlost. Dále jsou zahrnuty funkce pro měření pozice, čtení a potvrzování chybových hlášení a nastavení parametrů. Pro řízení víceosých systémů lze použít funkce vačkových profilů nebo elektronické převodovky.

5.1 PROGRAMOVÁNÍ

V předešlé části textu byla popsána reprezentace fyzické řízené osy v prostředí NC koncept, kde je na tuto osu pohlíženo jako na datovou strukturu. Pro adresaci této struktury je použito unikátní identifikační číslo ID, které je přiděleno při inicializaci dané osy. Toto ID je následně využíváno k adresaci v řídicích funkcích.

5.1.1 „ncaccess“ funkce

Tato funkce je provedena pouze jednou a to při inicializaci každého objektu řízené osy. Po úspěšné inicializaci vrací tato funkce unikátní číslo ID reprezentující adresu příslušné osy, sloužící k přístupu. Pro inicializaci datové struktury osy lze mimo funkce „ncaccess“ použít i funkci „ncalloc“, tato funkce se používá v případě, kdy není deklarována deployment tabulka, do funkce jsou předávány parametry, které jsou za použití „ncaccess“ definované v deployment tabulce, jako typ spojení, číslo modulu a jiné. Důsledkem je nižší přehlednost a menší flexibilita. V demonstrační úloze je použita funkce „ncaccess“, která vrací adresu alokované datové struktury pro příslušnou osu.

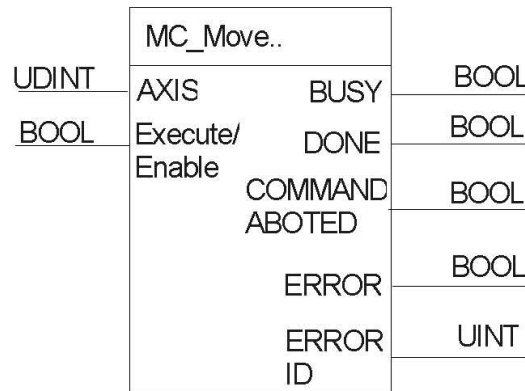
status = ncaccess(nc_sw_id, nc_obj_name, adr(nc_object))

Vstupní parametry:		
nc_sw_id	UINT	NC Software : ncACPMC10, specifikace produktu
nc_obj_name	UDINT	Zástupné jméno definované v deployment tabulce
Výstupní parametry:		
nc_object	UDINT	Specifické ID, reference na deklarovaný objekt
Status	UINT	Stav akce, úspěšná vrací 1, nebo vrací číslo chyby

Tabulka 5.1 Parametry funkce „ncaccess“ [4]

5.1.2 Funkční bloky

Všechny funkční bloky definované v knihovně ACP10MC mají stejnou strukturu.



Obrázek 5.1 Funkční blok [3]

Vstupem předávaným funkci při jejím volání jsou parametry:

- AXIS - adresa osy, nad kterou se má funkce vykonat, adresa je předaná ve formě ID čísla, deklarovaného při inicializaci
- Execute/Enable - příkaz pro vykonání, tento příkaz se provede pouze jednou nebo je prováděn opakovaně, dokud je na vstupu hodnota true, tato vlastnost je specifická pro jednotlivé funkce

Funkce vrací stav vykonání:

- BUSY – funkce je vykonávána
- DONE – funkce byla vykonána úspěšně
- COMMAND ABOTED – funkce byla zrušena jinou funkcí
- ERROR – chyba při volání funkce
- ERROR ID – vrací číslo vyskytnuvší se chyby, podle kterého ji lze identifikovat v nápovědě

5.1.3 „naction“ funkce

„naction“ je funkce volaná v cyklické části programu, slouží k provedení akce nad objektem, tyto parametry jsou předány při jejím volání, funkce vrací stav vykonané akce případně chybu.

status = naction(nc_object, subject, action)

Vstupní parametry:		
nc_object	UDINT	Adresa objektu z funkce „naccess“
Subject	UINT	Subjekt z datové struktury NC object
Action	UINT	Akce pro vykonání
Výstupní parametry		
Status	UINT	Stav akce

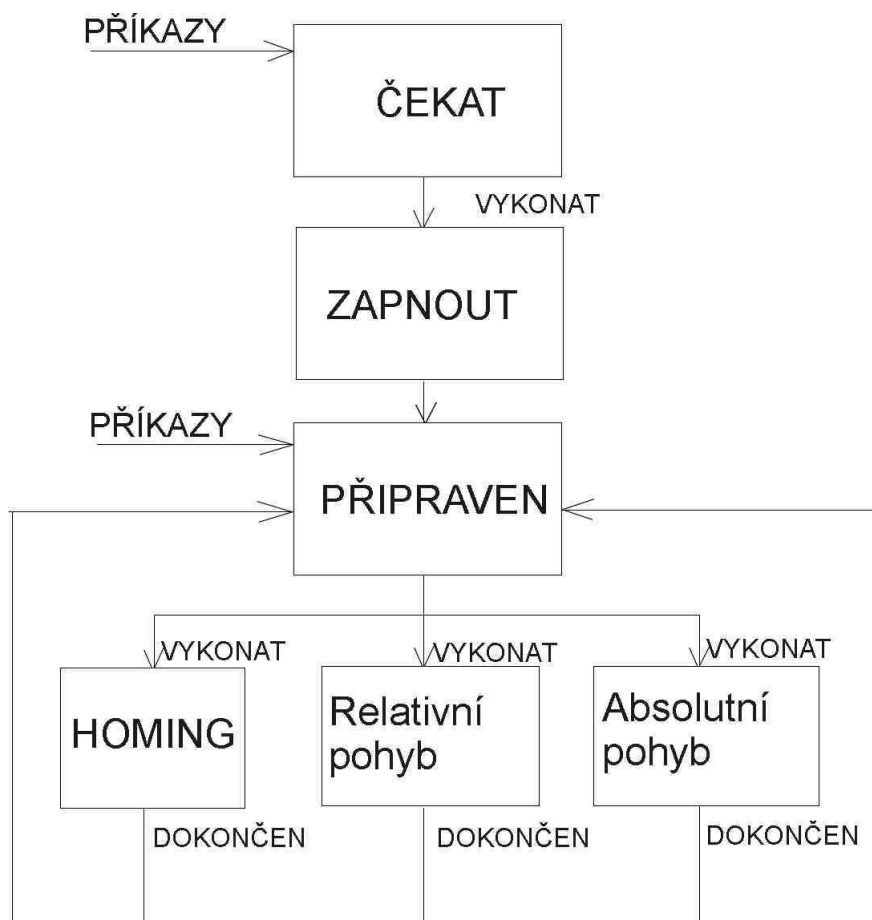
Tabulka 5.2 Parametry funkce „naction“ [4]

Ukazatel „nc_object“ obsahuje adresu datové struktury reprezentující příslušnou osu, parametry této struktury jsou logicky členěny do bloků, subjektů, tyto subjekty jsou controller, encoder interface a jiné. Tyto subjekty mohou být vykonány různými akcemi s rozdílnou prioritou, jako například inicializace, start. Stav vykonávané akce je vrácen ve výstupním parametru. Je-li akce zpracována NC managerem, funkce vrací stav ncOK, nedojde-li k okamžitému vykonání, vrací stav ncACTIVE, dojde-li k chybě, je navrácen stav ncERROR a je zobrazeno i číslo příslušného erroru.

5.2 STRUKTURA PROGRAMU

Jakýkoliv program pro obsluhu jedno i víceosého systému musí mít stejnou základní strukturu a musí obsahovat minimálně dvě funkce popsané výše. Program lze rozdělit do dvou částí, jednak je to část inicializační, která se provede pouze jednou při startu programu. Tato část musí obsahovat funkci, která alokuje paměťový prostor pro všechny deklarované osy, za použití funkce „ncalloc“ nebo „naccess“. Došlo-li k naalokování paměťového prostoru je možné s danou osou pracovat v cyklické části programu. Cyklická část programu, je prováděna v každém cyklu běhu PLC a obsahuje funkce pro řízení dané osy. V této části se používá funkce „naction“. Jednotlivé akce musí následovat v určitém pořadí, například akce pro vykonání pohybu musí následovat až po akci pro zapnutí servozesilovače. Další podmínkou pro korektní chování programu je nutnost definovat podmínky za jakých lze provést následující akci. Pro toto řízení běhu programu lze s úspěchem využít výstupních parametrů při volání funkce „naction“. Knihovna ACP10_MC obsahuje funkční bloky, pomocí nichž lze vystavět přehlednou stavovou strukturu pro sekvenční provádění

jednotlivých kroků. Mimo běžných bloků obsahuje knihovna i blok pro správu chybových stavů.



Obrázek 5.2 Stavový diagram úlohy [3]

5.2.1 Motion control sample project

Mimo možnosti samostatně si vytvořit kompletní program z jednotlivých funkčních bloků, jak bylo popsáno výše, se dnes téměř výhradně používá již napsaný program, který je součástí automation studia. Program obsahuje stavovou strukturu se základními stavy, které obsahují základní pohyby pro řízení osy pomocí servozsilovače. Program je psaný ve třech programovacích jazycích a to strukturovaný text (ST), Ladder diagram (LD) a ANSI C (C), dále jsou dodrženy konvence standardu PLCopen, součástí musí být knihovna ACP10_MC. K inicializaci datové struktury osy je použita funkce „*ncaccess*“, pro to je nezbytné použít deployment tabulku. Program lze libovolně upravovat a doplňovat dalšími funkcemi, zejména specifické funkce pro servozsilovače B&R, které nejsou standardně implementovány. Výhodou je stabilita a funkčnost algoritmu, zároveň jsou ošetřeny všechny havarijní stavy, ve

kterých se může systém vyskytnout. Samozřejmostí je možnost použití inicializačních modulů INIT param modul. V programu lze pracovat s dvěma reálnými osami pro jejich synchronizaci lze použít funkce elektronické převodovky nebo vačkové profily.

5.2.2 Použité funkce

Výčet nejčastěji používaných funkcí a procedur v Motion control sample project.

Datová struktura, která obsahuje parametry a příkazy pro základní pohyby:

```
_GLOBAL AxisBasic_typ gAxisBasic;
```

Pro inicializaci je použita funkce:

```
AxisIErrorNcAccess = ncaccess(ncACPIOMAN, "osal", &(AxisIObj));
```

Parametry "osal" je zástupné jméno inicializované osy v deployment tabulce.

UDINT &(AxisIObj) ukazatel na adresu alokované struktury

Po korektní alokaci přechází program do stavu *STATE_WAIT*, je-li nastavena hodnota proměnné *gAxisBasic.Power == 1*; dojde k zapnutí ACOPOSu, nevyskytnou-li se chyby, přechází program do stavu *STATE_READY* z něhož jsou volány další stavy při zápisu hodnoty true pro vykonání do příslušných proměnných a po skončení akce se do tohoto stavu program vrací. Ve stavu *STATE_READY* je možné měnit parametry pohybu jako je cílová vzdálenost, rychlost, zrychlení, zpomalení a směr.

```
gAxisBasic.ParaPosition;
```

```
gAxisBasic.ParaVelocity;
```

```
gAxisBasic.ParaAcceleration;
```

```
gAxisBasic.ParaDeceleration;
```

```
gAxisBasic.ParaDirection;
```

Start akce se provede zápisem hodnoty true do příslušné proměnné:

```
gAxisBasic.MoveAbsolute = 1;
```

Není-li akce blokována jinou s vyšší prioritou nebo chybou přejde program do stavu pro vykonání volané akce, nastaví se její parametry a akce se začne provádět. Je-li akce skončena, přechází automat zpět do stavu *STATE_READY*, podle stavového diagramu úlohy, jak je zobrazena na obrázku 5.2. Program obsahuje řadu dalších akcí a také stav při vyskytnuvší se chybě.

5.2.3 Speciální funkce

Program basic obsažený v Motion Control Sample Project obsahuje funkce podle standartu PLCopen. Další specifické funkce pro servozesilovače ACOPOS, jako je čtení a zápis

parametrů regulátorů a dalších vnitřních parametrů, nejsou implicitně obsaženy a programátor je musí dopsat. Tyto parametry jsou řízeny NC operation systemem, který komunikuje s uživatelským programem prostřednictvím NC manager. Tyto funkce jsou obsaženy v knihovně ACP10_MC a mají stejnou strukturu a stejná pravidla pro použití jako běžné funkce. Pro rozlišení se používá pojmenování „MC_BR...“. Jednotlivé parametry v servozesilovači mají přiřazeno unikátní číslo ParID, to se používá k adresaci příslušného parametru. Tímto rozšířením jsou pokryty veškeré požadavky na obsluhu.

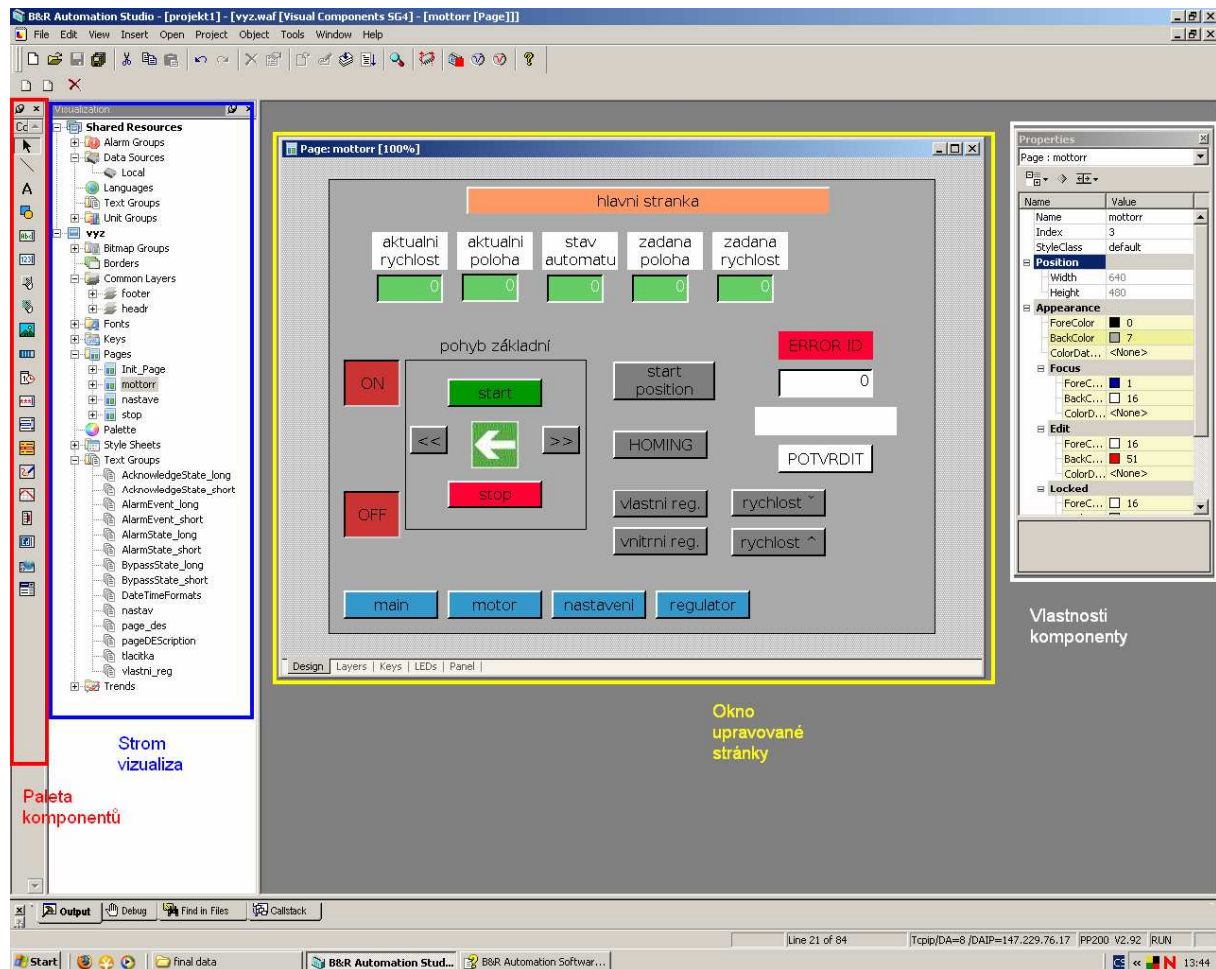
Využití specifických funkcí je například v demonstrační úloze, kde je vyřazen z činnosti vnitřní regulátor v servozesilovači a je nahrazen regulačním algoritmem běžícím na PLC, zde je nutné cyklicky číst a zapisovat do proměnných ve struktuře kaskádně zapojených regulátorů, které by byly běžnými funkcemi nepřístupné.

6. VIZUALIZACE

Pro obsluhu zařízení je nutné vytvořit spojení mezi strojem a jeho obsluhou, při použití automatického řízení nemusí hrát toto spojení mezi člověkem a strojem primární roli, ale je nezbytné pro správu zařízení. První zobrazovací a nastavovací prvky v počátcích automatického řízení, zastoupené signálkami a tlačítky, jsou dnes s rozvojem výpočetní techniky nahrazeny sofistikovanými systémy schopnými zobrazovat nezbytné informace v grafické podobě. K zadávání příkazů slouží jak tlačítka, klávesnice tak i dotykové panely. Produkty firmy B&R obsahují vizualizační komponenty v celé škále využití od nejspodnější úrovně procesního řízení, kde jsou převážně zastoupeny textové řádkové zobrazovače, nebo jednoduché grafické panely až po řízení technologii, strojů a systému SCADA, kde se uplatňují velké grafické dotykové panely, nebo vizualizační prostředí běžící na průmyslovém PC.

Vývojový nástroj automation studio obsahuje komponentu, která umožňuje snadnou tvorbu vizualizačních aplikací pro použití na operátorském panelu, anebo při využití PLC jako serveru, je možné zpřístupnit vizualizaci a ovládání pomocí běžného internetového prohlížeče.

Po nainstalování zobrazovače do vytvářeného projektu se vytvoří v softwarové části komponenta, umožňující snadnou tvorbu vizualizace. Součástí této komponenty je ve volitelném umístění paleta používaných nástrojů, které jsou přetažením vloženy do vytvářené stránky. Takto je možné přidávat textová, numerická pole, pro zobrazení i zadávání hodnot. Virtuální tlačítka, je-li použit dotykový panel a jiné. Dále je zobrazen strom s výčtem všech objektů, zde jsou seřazeny do společných bloků všechny vytvořené stránky, seznamy tlačítek, textů, proměnných a jiné. Hlavní plochu zabírá vytvářená stránka. Další volitelnou komponentou, jež je možné zobrazit, představuje tabulka s editací vlastností zvoleného objektu. Provázání vizualizace s projektem se děje pomocí proměnných. Tabulka Data Source obsahuje seznam všech použitých proměnných v celém projektu, po změně těchto proměnných je nutné ji vždy aktualizovat. Veškeré texty lze vytvořit ve více jazykových verzích a jednoduše mezi nimi přepínat. To umožňuje vytvořit multijazykové verze programu. Na každou stránku lze přidat více vrstev a takto vytvořit záhlaví, zápatí stejné pro všechny stránky.



Obrázek 6.1 Prostředí pro tvorbu vizualizační aplikace

6.1 ZOBRAZOVACÍ ZAŘÍZENÍ

Ve vytvořené aplikaci je na pozici PLC použit Power Panel s dotykovou obrazovkou.

Power Panel PP 282:

- 10,4“ VGA colour TFT display s dotykovým panelem
- 44 funkčních kláves, 20 systémových kláves
- 1 aPCI slot; 64 MB SDRAM; 256 kB SRAM
- Compact Flash slot (type I); ETH 10/100
- RS 232; 2xUSB
- Ochrana IP 65 (přední strana)
- Napájení 24 V DC

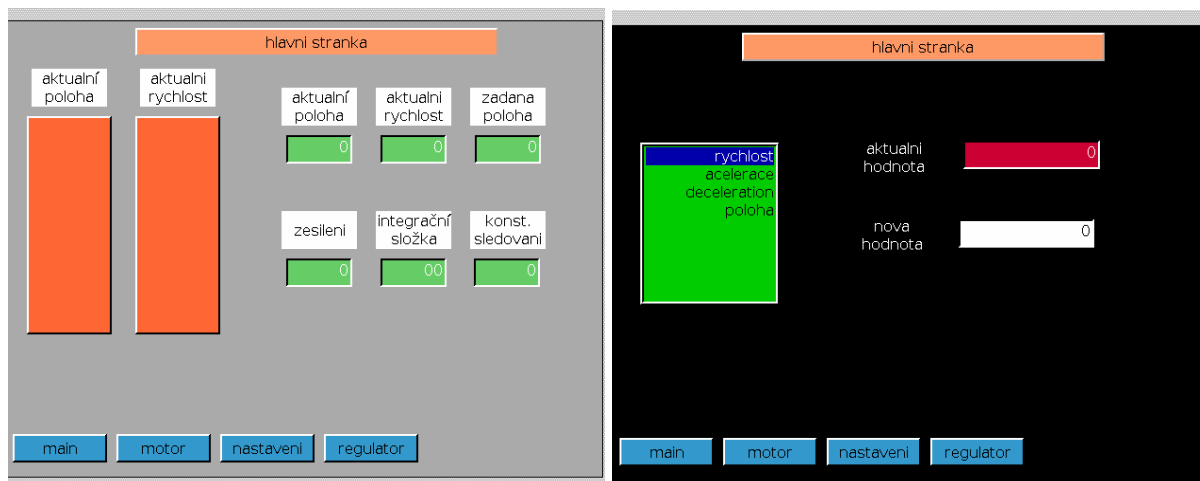


Obrázek 6.2 Power Panel [12]

6.2 PROGRAM PRO VIZUALIZACI

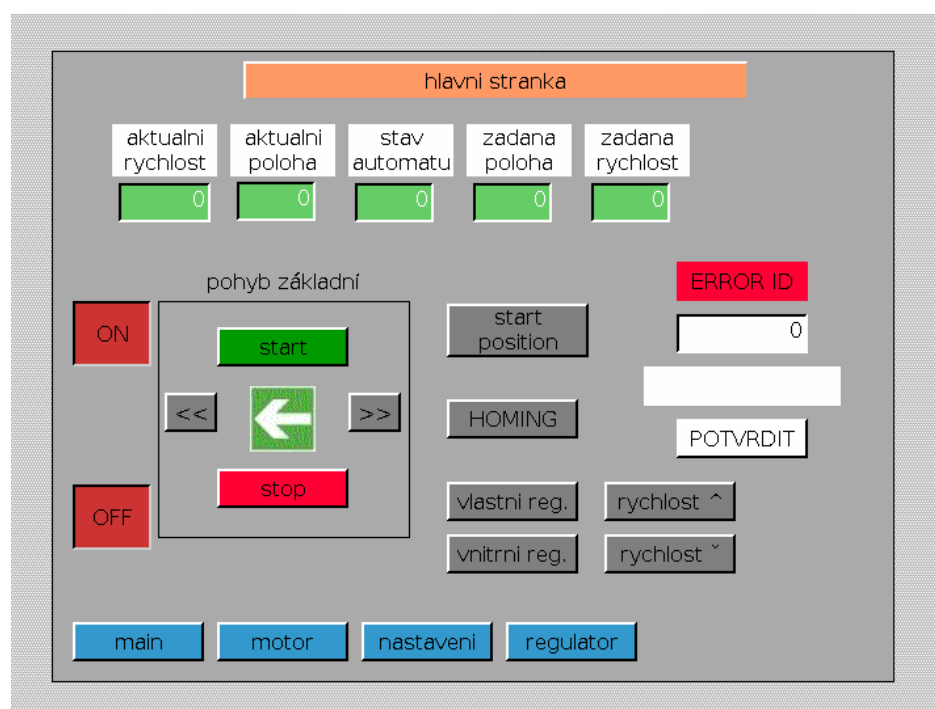
K ovládání servopohonu slouží Power Panel s dotykovou obrazovkou, skrze ni je realizováno ovládání a zobrazení stavu. Byly vytvořeny čtyři stránky, společné mají zápatí, kde se nacházejí virtuální tlačítka k přepínání mezi stránkami, přepnout lze i za pomoci fyzických tlačítek pod panelem. V záhlaví se zobrazuje název právě zobrazené stránky. Tlačítka a zobrazovače jsou provázány skrze globální i lokální proměnné s řídicím programem. K zadání parametrů slouží stránka *nastaveni*, tato obsahuje ListBox, kde je vybrán nastavený parametr a v numerickém poli lze zadat novou hodnotu. Po dotyku na nastavované komponenty se zobrazí zadávací pole jako Up/Down menu, nebo virtuální numerická klávesnice.

K zadávání parametrů vlastního PS regulátoru slouží stránka *regulator*. Tato stránka obsahuje numerická pole, jež zobrazují aktuální hodnoty, a po dotyku se zobrazí numerická klávesnice, která umožňuje tyto hodnoty měnit. Pro ilustraci schopností vizualizační komponenty jsou součástí stránky i dva sloupcové grafy znázorňující aktuální polohu a rychlost.



Obrázek 6.3 Stránka pro zadání parametrů PS regulátoru, stránka pro nastavení

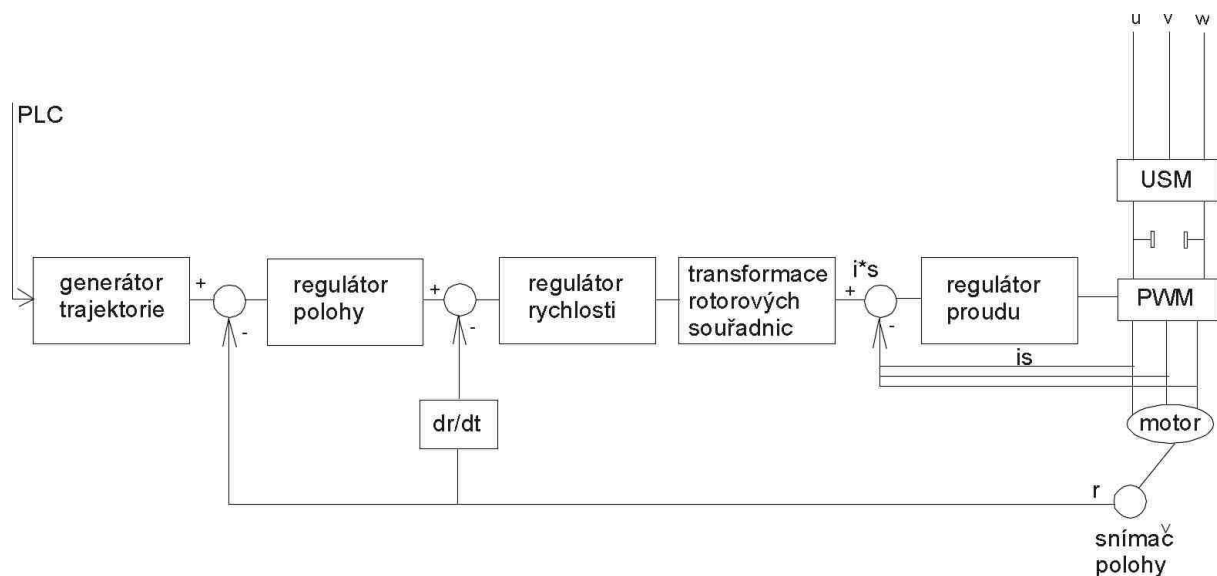
Stránka nazvaná *motor* obsahuje numerická pole informující o stavu motoru a dále virtuální tlačítka pro základní ovládání a přepínání mezi vnitřním regulátorem a PS regulátorem běžícím na PLC. Servozesilovač lze zapnout i vypnout pomocí tlačítka. Dále je možné určit směr otáčení indikovaného šipkou a zvolit ze dvou režimů regulace na zadanou polohu nebo rychlost. Vyskytnuvši se porucha je zobrazena identifikačním číslem, a je-li k dispozici i slovní popis, je také zobrazen, pro kvitování poruchy slouží tlačítko *POTVRDIT*. Při zapnutí servozesilovače jen nezbytné provést proceduru homing, jenž nuluje aktivní pozici, nebo uvádí hřídel do definované výchozí pozice, pro tuto proceduru slouží tlačítko *HOMING*.



Obrázek 6.4 Stránka pro ovládání pohonu

7. ACOPOS CONTROL CONCEPT

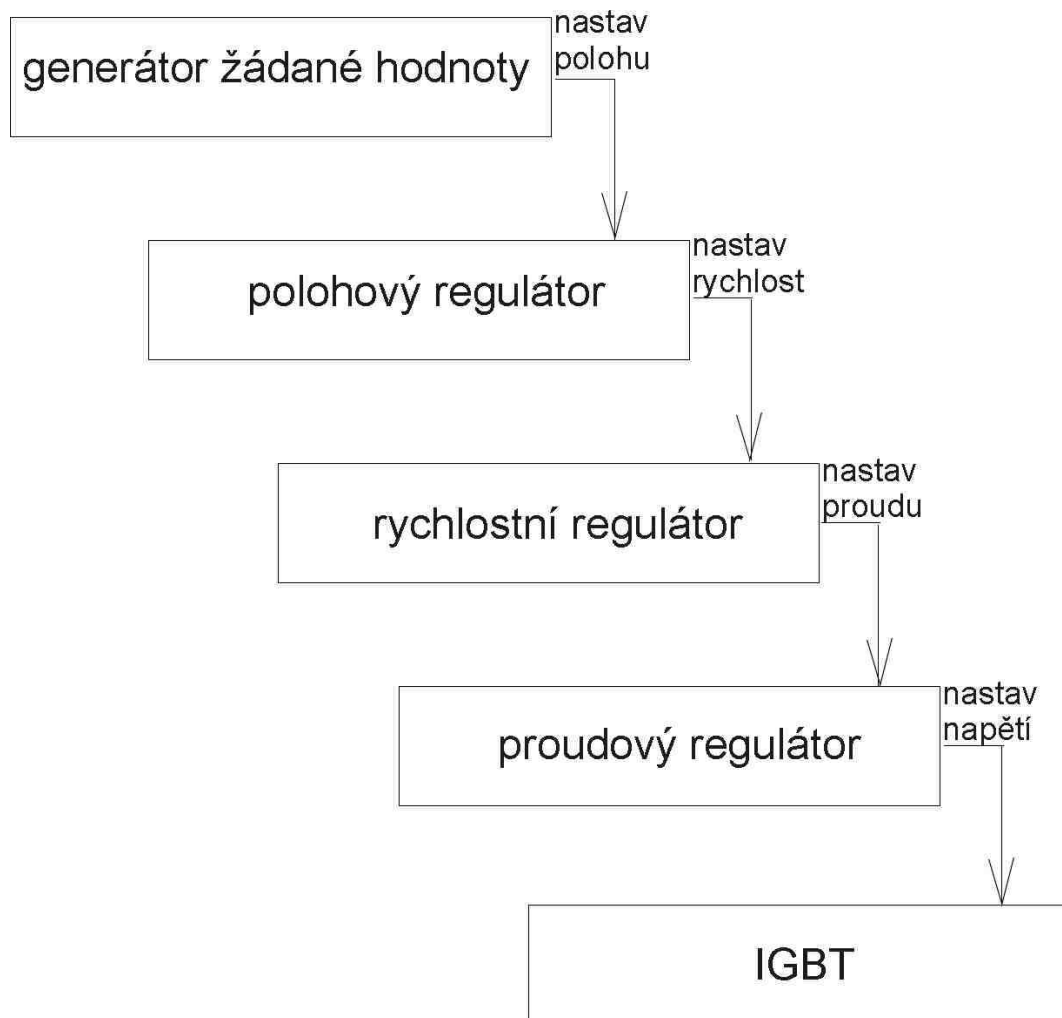
Požadavkem na servopohon je urazit požadovanou vzdálenost nebo udržovat zvolenou rychlost s maximální přesností a s minimální prodlevou. Ke splnění těchto úkolů je zapotřebí zpětná informace o aktuálních parametrech, tu poskytuje měřicí člen resolver, nebo jiný typ. Změřená aktuální hodnota je porovnávána s hodnotou žádanou, jejich rozdíl je vyhodnocen a příslušně upraven v regulátoru, který veličinou na svém výstupu ovládá akční člen v tomto případě frekvenční měnič pro napájení motoru. Volba vhodného regulátoru a jeho správné nastavení je kritickým prvkem, jenž významně ovlivňuje celkový výsledek.



Obrázek 7.1 Celková regulační smyčka [8]

7.1 KONCEPCE KASKÁDNÍHO ŘÍZENÍ

Servozesilovače ACOPOS obsahují tři diskrétní regulátory v kaskádním zapojení a generátor žádané hodnoty, který nastavuje žádanou hodnotu pro polohový regulátor podle zadaných parametrů pohybu od uživatelského programu. Akční zásah od nadřazeného regulátoru vytváří žádanou hodnotu pro následující podřízený regulátor. Polohový, rychlostní i proudový regulátor jsou typu PI s dalšími rozšířeními lišícími se podle funkce regulátoru. Servozesilovač obsahuje vnitřní logiku, která počítá regulační algoritmy. Pro realizaci zpětné vazby je volen snímač otáček implementovaný na hřídeli motoru, případně doplněn o snímač na hnané soustavě. Rychlostní a polohový regulátor lze vyřadit z činnosti a nahradit je vlastním regulačním algoritmem běžícím na PLC.



Obrázek 7.2 Kaskádní řazení regulátorů [5]

7.2 GENERÁTOR ŽÁDANÉ HODNOTY

Žádaná hodnota pro polohový regulátor a od něj odvezené další regulátory se získává z generátoru, který pracuje s cyklickým časem 400 μ s. Úkolem generátoru je vytváření profilů, po kterých probíhá vykonávaný pohyb. Vlastnosti těchto profilů závisí na parametrech pohybu, jako je cílová vzdálenost, hodnoty zrychlení a zpomalení, aktuální poloha a další. Tyto parametry jsou nastavovány v bloku Basic Movement Parameters.

Na začátku nebo na konci pohybu dochází ke skokové změně zrychlení, nazývané „jolts“. Tento skok způsobuje zvýšení nároku na velikost krouticího momentu, který musí být motorem poskytnut, to motor zbytečně zatěžuje a způsobuje nežádoucí vibrace. K omezení je generátor žádané hodnoty vybaven „jolts filtrem“ s nastavitelnou hodnotou „t_jolt“ v rozpětí 0 až 0,2 sekundy. Nastavení nenulové hodnoty způsobí změnu ve tvaru křivky zrychlení, kde nedochází ke skoku na maximální hodnotu, ale zrychlení je postupně zvyšováno se strmostí

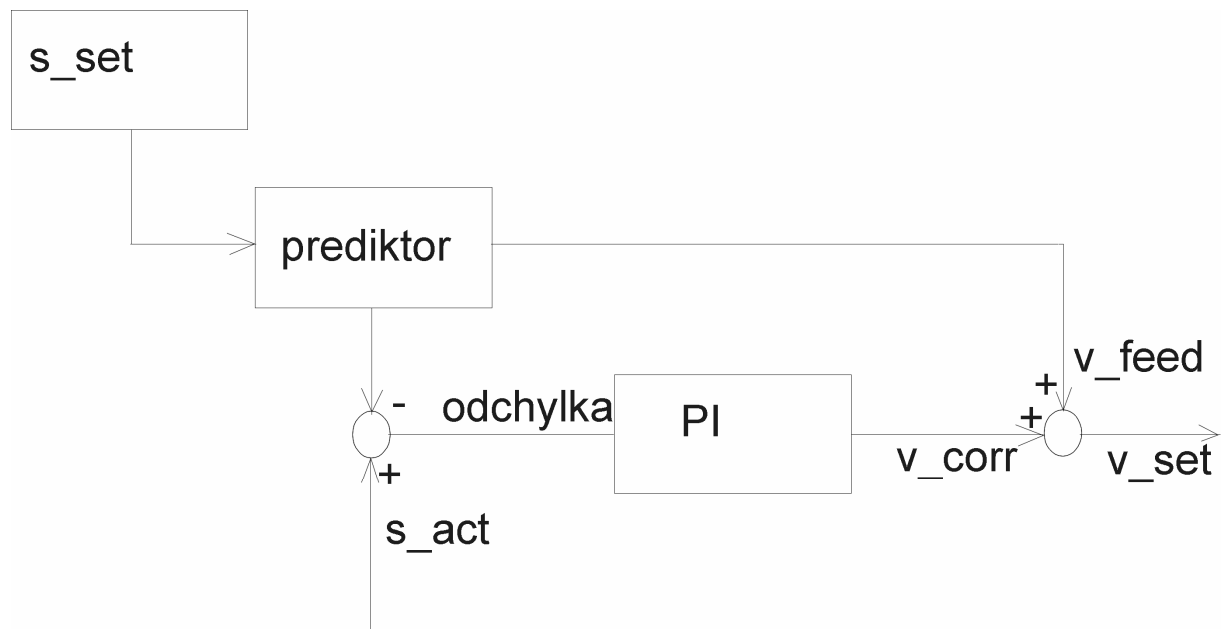
danou nastavením parametru „t_jolt“. Maximální hodnoty zrychlení je dosaženo za čas „t_jolt“ v sekundách.

Nastavení v bloku Limit Values.

7.3 POLOHOVÝ REGULÁTOR

Prediktivní polohový regulátor obdrží žádanou hodnotu z generátoru žádané hodnoty (s_{set}) a aktuální polohu (s_{act}) ze snímače na hřídeli motoru prostřednictvím zásuvné karty obsahující interface ke snímači. Vzniklá regulační odchylka je zpracována a vypočítaný akční zásah je poslán jako žádaná hodnota pro rychlostní regulátor.

Polohový regulátor je typu PI s anti-windupem a prediktivní dopředno vazbou. Jak P složka tak i I složka mohou být omezeny na maximální nastavenou hodnotu.



Obrázek 7.3 Polohový regulátor [5]

Zařazení dopředné vazby přidává do PI regulátoru přídavné zpoždění, se kterým je zpracována regulační odchylka, na výstupu z regulátoru je sumována složka prediktivní (v_{feed}) daná žádanou hodnotou z generátoru a zpracovaná regulační odchylka s dopravním zpožděním (v_{corr}). Výsledkem je žádaná rychlost pro rychlostní regulátor (v_{set}). Přidané dopravní zpoždění je možné nastavovat. Zařazení dopředné vazby zlepšuje dynamické chování systému.

Pro nastavení parametrů slouží blok Controller Position . Pro prvotní nastavení jednotlivých konstant regulátoru lze použít následujících vzorců, které mají empirický základ a jsou k nalezení v dokumentaci [5].

- Suma jednotlivých časových konstant

$$T_{\sigma_p} = T_{interpol} + 4 \times T_{\sigma_v} + T_{dead_p} \quad (7.3.1)$$

$T_{interpol}$ = mrtvá doba interpolace (0,0001 s)

$4 \times T_{\sigma_v}$ = časová konstanta rychlostní smyčky

T_{dead_p} = mrtvá doba snímače (0,0001 s)

- Zesílení

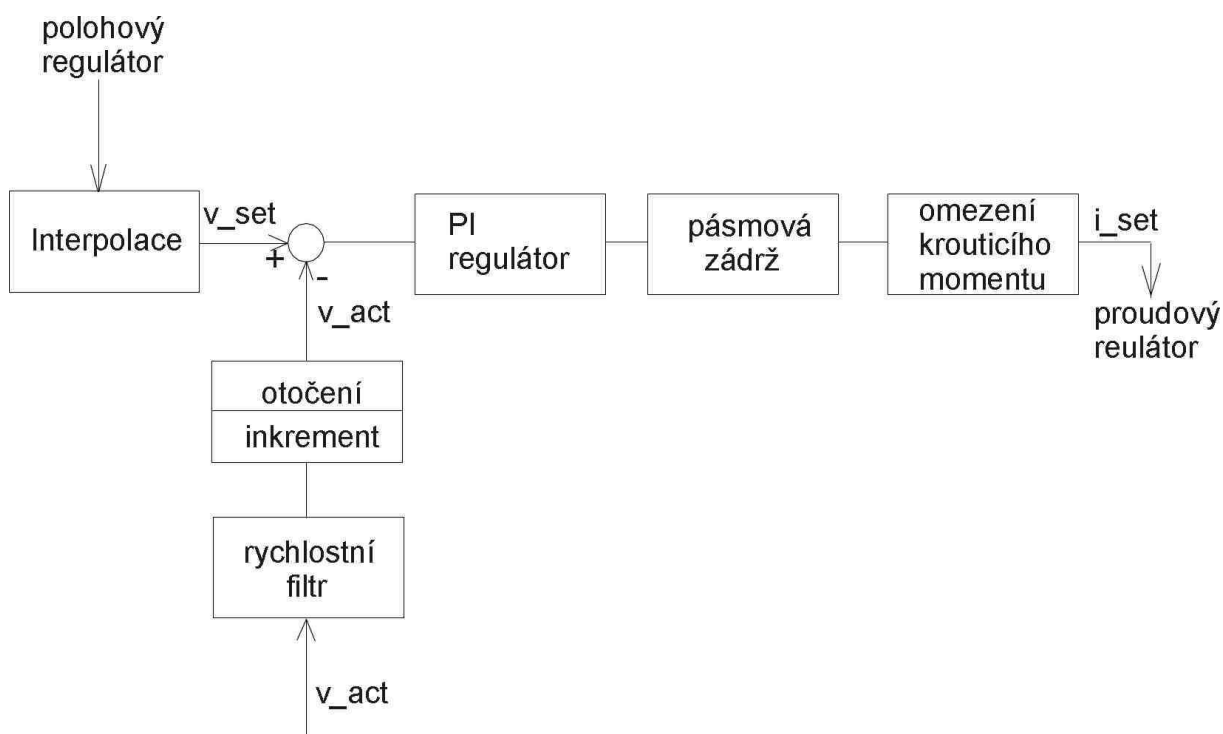
$$kv = \frac{1}{2 \times T_{\sigma_p}} \quad (7.3.2)$$

- Integrovaná konstanta

$$tn = 4 \times T_{\sigma_p} \quad (7.3.2)$$

7.4 RYCHLOSTNÍ REGULÁTOR

Rychlostní regulátor má za úkol regulaci rychlosti otáček motoru na základě žádané rychlosti nastavené nadřazeným polohovým regulátorem. Před zpracováním je nutné tento údaj interpolovat z důvodu rozdílné rychlosti s jakou jednotlivé regulátory pracují. Rychlostní regulátor pracuje s cyklickým časem 200 μ s. Žádaná rychlost je porovnávána s aktuální rychlostí vypočítanou jako diference polohy, snímanou enkodérem na hřídeli motoru, za čas. Vzniklá regulační odchylka je vyhodnocena PI regulátorem s anti-windupem. Výstup z regulátoru slouží pro nastavení žádané hodnoty regulátoru proudu.



Obrázek 7.4 Rychlostní regulátor [5]

Aktuální rychlost (v_{act}) je snímána jako množství inkrementů za sekundu. Signál je před zpracováním filtrován rychlostním filtrem typu dolní propust, z důvodu potlačení vyšších rušivých frekvencí. Poté je hodnota rychlosti převedena z jednotek udávaných v počtu inkrementů za sekundu do jednotek ve tvaru otáček za sekundu a tento údaj vstupuje do regulátoru jako aktuální hodnota (v_{act}).

Akční zásah vystupující z PI regulátoru může být filtrován pásmovou zadržicí, která odstraňuje vysoké frekvence šumu a zároveň potlačuje rezonanční frekvence, které se mohou v mechanickém systému objevit.

Z důvodu ochrany servozesilovače a motoru před přetížením vzniklým v důsledku požadavku na zvýšení krouticího momentu nad bezpečnou mez je zařazen blok Torque limiter (omezení krouticího momentu), který omezuje vzniklé proudové špičky na přípustnou mez.

Pro nastavení parametrů slouží blok Controller Speed. Také pro rychlostní regulátor lze použít empirických vztahů k výpočtu jednotlivých konstant.

- Časová konstanta proudového regulátoru

$$T_I = 2 \times \left(0,000075 + \frac{1}{2 \times \text{frekvencePWM}} \right) \quad (7.4.1)$$

- Suma všech časových konstant

$$T_{\sigma_v} = T_I + T_{dead_v} + T_{filter} \quad (7.4.2)$$

$$T_{dead_v} = 0,000175 \text{ s (mrtvá doba enkodéru)}$$

$$T_{filter} (\text{„t_filter“ parametr}) = \text{časová konstanta rychlostního filtru}$$

- Zesílení

$$k_V = \frac{J \times \sqrt{2} \times \pi}{T_{\sigma_v} \times k_t} \quad (7.4.3)$$

$$J = \text{celkový moment soustavy (} J_{motor} + J_{brzda} + J_{břemeno} \text{)}$$

$$k_t = \text{konstanta krouticího momentu motoru [Nm/A]}$$

- Integrační konstanta

$$k_t = 4 \times T_{\sigma_v} \quad (7.4.4)$$

7.5 PROUDOVÝ REGULÁTOR

Stejně jako předešlé je i proudový regulátor typu PI. Jeho parametry se nastavují automaticky podle parametrů motoru a specifických parametrů ACOPOSu. Tento regulátor řídí tranzistory IGBT střídače, jenž napájí motor pulsně šířkově modulovaným signálem. Cyklický čas, s nímž regulátor pracuje je závislý na frekvenci PWM. Pro konkrétní typ

použitého servozsilovače ACOPOS 8V1022.00-2 je PWM frekvence 20 kHz, čemuž odpovídá cyklický čas 50 μ s.

8. NASTAVENÍ REGULÁTORŮ

Součástí projektu byla snaha o optimální nastavení regulátorů v servozesilovači. Dále je implementován vlastní regulační algoritmus na PLC, který nahrazuje vnitřní poziční regulátor, nastavený pro optimální chování.

Regulovanou soustavu tvoří synchronní motor, na hřídeli je upevněn malý hliníkový kotouč, jeho setrvačné působení je malé.

Nastavení konstant regulátoru se provádí zápisem do příslušné kolonky parametrické tabulky. Změny se projeví až po kompilaci a nahrání do servozesilovače. Nevýhodou je značná časová náročnost nutná pro nahrání každé změny. Velkou úsporou času by poskytovala možnost měnit parametry regulátorů za chodu programu nějakou funkcí. Standart PLCopen tyto funkce nepodporuje, ale některé specifické funkce „MC_BR_“ slouží ke čtení a zápisu vnitřních parametrů identifikovaných unikátním ID číslem. Tyto funkce lze využít jak pro čtení a zápis konstant regulátorů tak jsou nezbytné pro implementaci vlastního regulačního algoritmu, kdy je zapotřebí číst aktuální hodnotu a zapisovat žádanou hodnotu pro podřízený regulátor v kaskádní struktuře. V programu jsou využívány čtyři typy funkcí:

- „MC_BR_ReadParID“
- „MC_BR_WriteParID“
- „MC_BR_InitCyclicRead“
- „MC_BR_InitCyclicWrite“

První dvě funkce slouží k jednorázovému přečtení, nebo zápisu, druhé dvě slouží k cyklickému čtení nebo zápisu v každém cyklu běhu programu. Struktura práce s funkcemi jsou pro oba typy funkcí rovnocenné. Při práci s funkcemi je nutné vytvořit proměnné datového typu struktura, které obsahují proměnné se zadanými parametry.

```
_LOCAL MC_BR_InitCyclicRead_typ MC_BR_InitCyclicRead_actV;
```

```
MC_BR_InitCyclicRead_actV.Axis = Axis1Obj;
```

- Adresa dané osy, pro kterou je funkce volána

```
MC_BR_InitCyclicRead_actV.ParID = act_speed;
```

- ParID unikátní identifikátor parametru

```
MC_BR_InitCyclicRead_actV.DataType = ncPAR_TYP_REAL;
```

- Datový typ parametru

MC_BR_InitCyclicRead_actV.DataAddress = (UDINT)& act_v;

- Adresa pro ukládání/čtení hodnoty

MC_BR_InitCyclicRead(&MC_BR_InitCyclicRead_actV);

- Volání funkce pro provedení akce

if (MC_BR_InitCyclicRead_actV.Done == 1) MC_BR_InitCyclicRead_actV.Execute = 0;

- Funkce vrací stav provedení *Done*, je-li provedena, nebo *Error* při chybě.

Funkce se volá zápisem true do proměnné *Execute*

MC_BR_InitCyclicRead_actV.Execute = 1;

Proměnná *Execute* musí být po obdržení zprávy o stavu akce změněna na *false*, opakované volání v každém cyklu by mohlo způsobit zahlcení přenosové linky.

Potřebný identifikátor ID parametru je k nalezení v automation help, nebo v dokumentaci.

Výčet použitých parametrů

ID	jméno	jednotka	datový typ	status
rychlostní regulátor				
251	aktuální rychlost	1/s	REAL	RD
329	nastavená rychlost	Jednotky/s	REAL	RD, WR
253	zesílení regulátoru kv	As	REAL	RD, WR
255	integrační konst. regulátoru tv	s	REAL	RD, WR
polohový regulátor				
111	aktuální poloha	jednotka	DINT	RD
92	aktuální rychlost	jednotky/s	REAL	RD
113	nastavená poloha	jednotky/s	DINT	RD
114	nastavená rychlost	jednotky/s	REAL	RD
100	integrační konst. regulátoru tp	s	REAL	RD, WR
101	zesílení regulátoru kp	1/s	REAL	RD, WR
112	odchylka	jednotky	REAL	RD
231	blokovat regulátor	-	UINT	RD, WR

Tabulka 8.1 Použité vnitřní parametry [7]

8.1 POUŽITÍ VNITŘNÍCH REGULÁTORŮ

Při nastavování vnitřních regulátorů se postupuje od nejspodnějšího v kaskádní struktuře výše. Jako první se nastaví proudový regulátor, jeho parametry se nastavují automaticky, podle parametrů motoru a specifických parametrů ACOPOSu. Dále je potřeba

nastavit rychlostní regulátor. Pro první návrh lze použít vztahy uvedené v kapitole 7.4, potřebné konstanty lze nalézt v dokumentaci.

Parametry motoru a servozsilovače:

Motor 8MSA3L.R0-30, pro $v = 3000 \text{ min}^{-1}$		Servozsilovač ACOPOS 8V1022.00-2	
K_T [Nm/A]	1,39	PWM [kHz]	20
J_M [kg.cm ²]	1,2	Cykl.čas [μs]	50
J_B [kg.cm ²]	0,18		

Tabulka 8.2 Parametry motoru a servozsilovače [9]

Vypočítané hodnoty podle rovnic kapitoly 7.4:

Rychlostní regulátor:

$$T_I = 2 \times \left(0,000075 + \frac{1}{2 \times \text{frekvencPWM}} \right) = 2 \times \left(0,000075 + \frac{1}{2 \times 20000} \right) = 0,0002[s] \quad (8.1.1)$$

$$T_{\sigma_v} = T_I + T_{dead_v} + T_{filter} = 0,0002 + 0,000175 + 0 = 0,000375[s] \quad (8.1.2)$$

$$kv = \frac{J \times \sqrt{2} \times \pi}{T_{\sigma_v} \times k_t} = \frac{0,00015 \times \sqrt{2} \times \pi}{0,000375 \times 1,39} = 1,176[As / rev] \quad (8.1.3)$$

$$kt = 4 \times T_{\sigma_v} = 4 \times 0,000375 = 0,0015[s] \quad (8.1.4)$$

Polohový regulátor:

$$T_{\sigma_p} = T_{interpol} + 4 \times T_{\sigma_v} + T_{dead_p} = 0,0001 + 4 \times 0,000375 + 0,0002 = 0,0018[s] \quad (8.1.5)$$

$$kv = \frac{1}{2 \times T_{\sigma_p}} = \frac{1}{2 \times 0,0018} = 277,78[s^{-1}] \quad (8.1.6)$$

$$tn = 4 \times T_{\sigma_p} = 4 \times 0,0018 = 0,0072[s] \quad (8.1.7)$$

Nastavení regulátorů podle těchto vypočtených hodnot se ukázalo jako naprosto nevyhovující. Konstanty regulátorů lze zadávat v parametrické tabulce, hodnoty jsou nahrány do servozsilovače spolu s řídicím programem, pro každou změnu by bylo tedy nutné znovu provést transfer dat do PLC a do servozsilovače, to je poměrně časově náročné. Pro online změnu parametrů bez nutnosti restartu lze použít funkci „MC_BR_WriteParID“. Jednotlivé parametry jsou přístupné pomocí čísla ID. Obdobnou funkcí „MC_BR_ReadParID“ lze parametry číst. Použití těchto funkcí šetří čas nutný pro ladění konstant regulátorů.

8.1.1 Nastavení rychlostního regulátoru

Rychlostní regulátor je nastaven s podmínkou stability a omezení kmitů aktuální rychlosti. Struktura PI je doplněna o filtr aktuální hodnoty rychlosti. Při použití nadřazeného pozičního regulátoru se sumační složkou, nemusí být sumační člen obsažen v rychlostním regulátoru. Ve zvoleném nastavení doplňuje chování proporcionální složky, ale nemá zásadní vliv na odezvu celé soustavy. Pro odstranění šumu vzniklého snímáním a přenosem signálu aktuální hodnoty je zařazen před vstup regulátoru filtr typu dolní propust s nastavitelným mezním kmitočtem.

Proporcionální složka	P [s ⁻¹]	0,8
Integrační složka	I [s]	27
Mezní perioda filtru	T _f [s]	0,002

Tabulka 8.3 Parametry rychlostního regulátoru

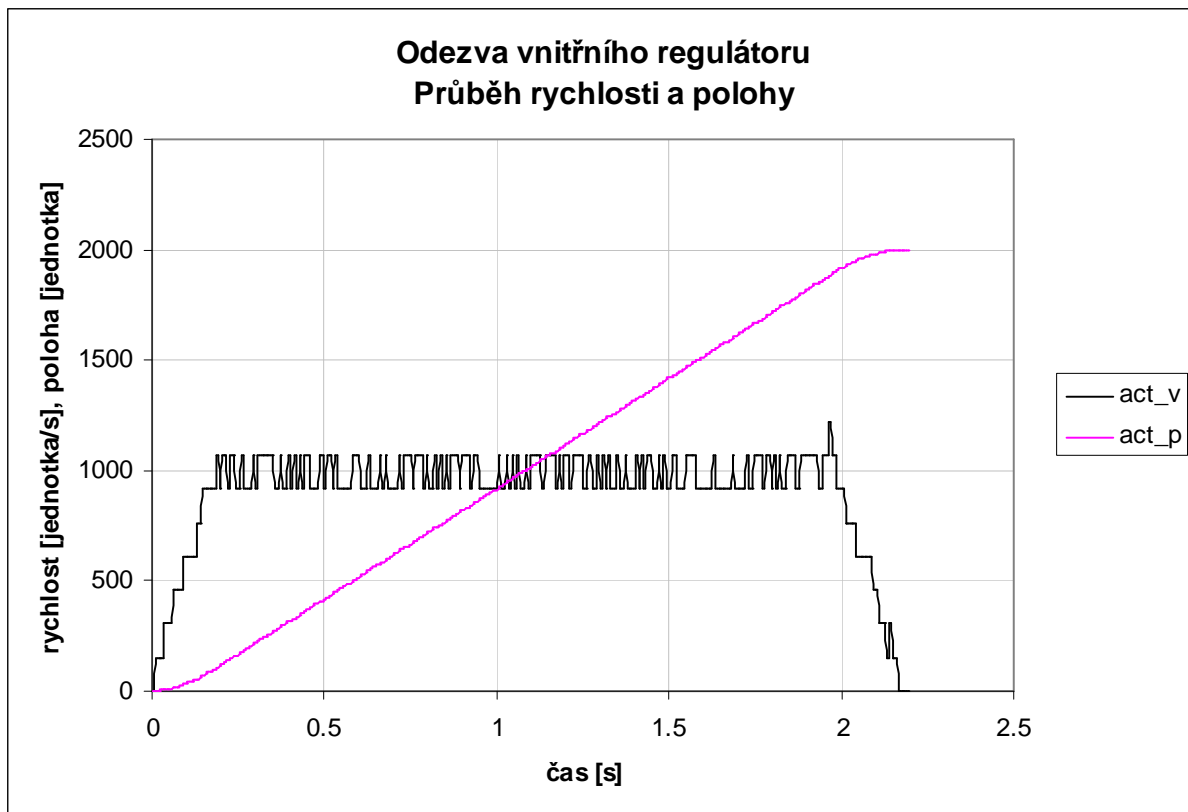
8.1.2 Nastavení polohového regulátoru

Primárním požadavkem bylo dosažení žádané pozice bez překmitu. Rychlost přechodného děje a tím i dynamika celé soustavy je závislá na nastavení maximálních otáček a maximálního zrychlení. Toto nastavení nesmí překročit maximální dovolené zatížení, které je motor schopen poskytnout a je třeba hodnoty volit s ohledem na připojenou technologii. Tvar přechodové charakteristiky dále ovlivňuje předřazený generátor žádané hodnoty, který generuje tvar průběhu charakteristiky, nastavením parametru lze omezit zrychlení na začátku a na konci přechodného děje, kdy by požadavek na skokovou změnu zrychlení mohl vyvolat nárůst proudu motorem nad dovolenou mez. Tato úprava poskytuje plynulejší průběh a tím šetří použitou technologii, důsledkem je mírné prodloužení doby trvání přechodného děje a mírná deformace přechodové charakteristiky.

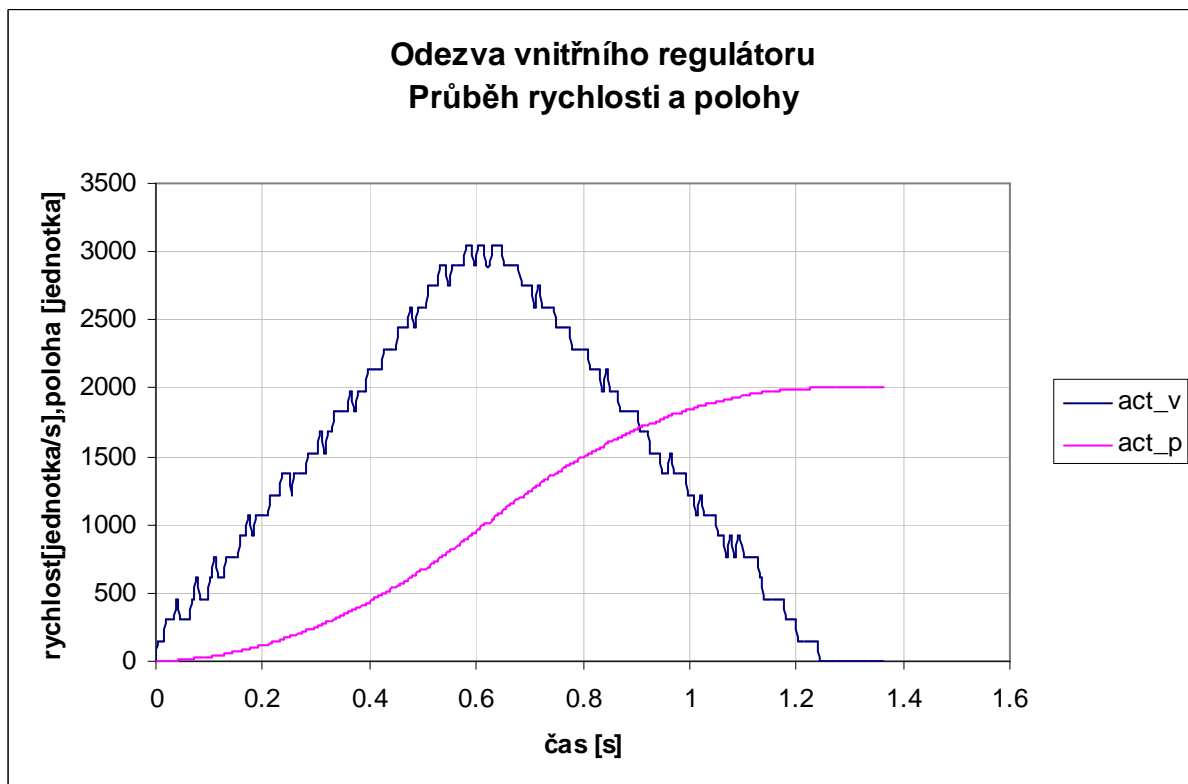
Mimo konstanty PI regulátoru lze nastavit i zpoždění pro použití dopředné vazby a omezení proporcionální a sumační složky.

Proporcionální složka [s ⁻¹]	65,25
Sumační složka [s]	0,03652
Celkové zpoždění [s]	0,002
Omezení proporcionální složky	10000
Omezení sumační složky	10000

Tabulka 8.4 Parametry polohového regulátoru



Obrázek 8.1 Odezva vnitřních regulátorů $v_{\max} = 1000$ jedn./s



Obrázek 8.2 Odezva vnitřních regulátorů $v_{\max} = 3000$ jedn./s

Obrázky 8.2 a 8.1 ilustrují závislost tvaru odezvy na změnu žádané polohy, při volbě různých hodnot maximální rychlosti a zrychlení. Pro oba obrázky je shodná hodnota cílové polohy 2000 jednotek. V obrázku 8.1 je volena maximální rychlost 1000 jednotek.s⁻¹ a zrychlení 1000 jednotek.s⁻², obrázek 8.2 ilustruje volbu maximální rychlosti 3000 jednotek.s⁻¹ a zrychlení 5000 jednotek.s⁻². Je vidět markantní rozdíl v čase trvání přechodného děje. Motor v součinnosti se servozesilovačem je schopen poskytnout trvale tyto hodnoty.

8.2 IMPLEMENTACE VLASTNÍHO POLOHOVÉHO REGULÁTORU

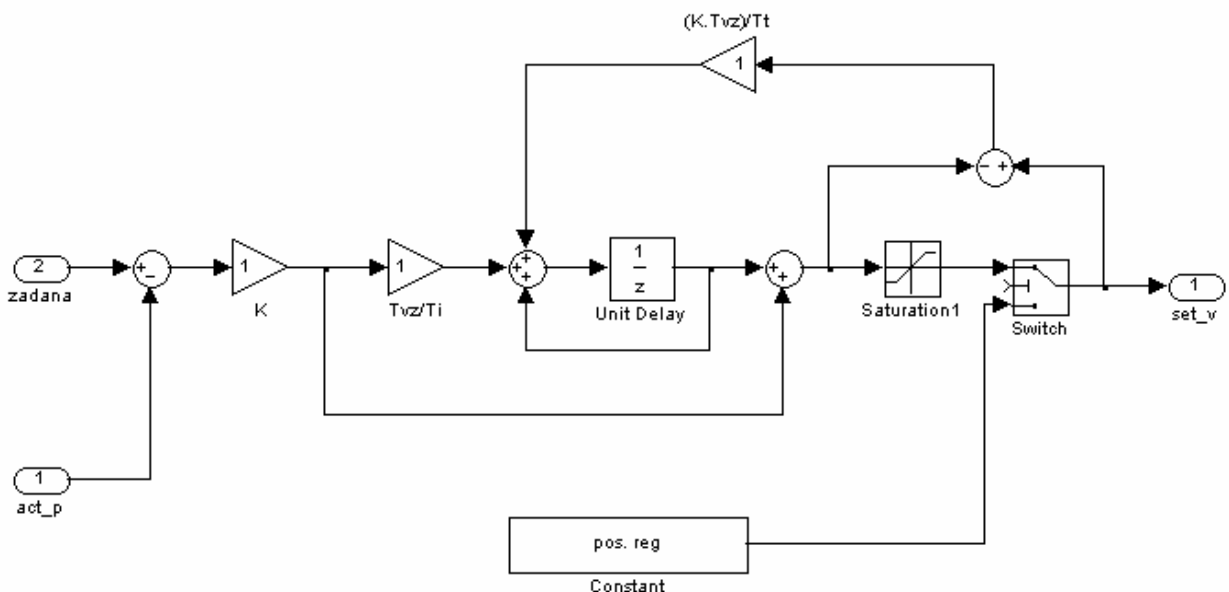
Implementace vlastního regulátoru vyžaduje zásah do vnitřní struktury regulátorů v servozesilovači. Tyto zásahy jsou uskutečněny funkcemi „MC_BR_WriteParID“ a MC_BR_InitCyclicWrite“.

aktuální hodnota je čtena z parametru *actual position*, přístupné pomocí ID 111, tato je ukládána do lokální proměnné *act_p*. Akční zásah vypočtený regulátorem je potřeba předat jako žádanou hodnotu podřízenému rychlostnímu regulátoru, to se děje zápisem do parametru *set speed*, přístupné pomocí ID 329. Tato vazba musí být spojena před zápisem do parametru *enable input*, přístupné pomocí ID 231, který vypíná vnitřní polohový regulátor. Pro realizaci beznárazového přepínání je nutné znát i žádanou hodnotu nastavenou z generátoru, při funkci vnitřního regulátoru. Tento parametr je možné číst pomocí *set position*, ID 113. Pro čtení aktuální hodnoty je nutné použít cyklického čtení realizovaného funkcí „MC_BR_InitCyclicRead“, pro zápis akčního zásahu je použita funkce „MC_BR_InitCyclicWrite“, obě funkce pracují s cyklickým časem daným periodou, s jakou pracuje NC Manager, tuto je možné nastavit. Při použití spojení pomocí ETHERNET POWERLINK, musí být cyklický čas nastaven na celistvý násobek frekvence s jakou je realizována komunikace, tj. 400μs. Pro zakázání funkce polohového regulátoru je volána funkce „MC_BR_ReadParID“, která je volána pouze jednou a která provádí změnu parametru *enable input*, jenž umožňuje vypnutí polohového regulátoru. Mezi funkcemi pro čtení aktuální hodnoty a zápisu akčního zásahu lze realizovat jakýkoliv řídicí algoritmus.

Pro porovnání vlastností byl zvolen PS regulátor s beznárazovým přepínáním a dynamickým omezením přebuzení sumační složky. Pro prvotní identifikaci regulované soustavy byla zvolena metoda Ziegler-Nichols, z důvodu zjednodušení jejího použití je implementován i samotný P regulátor, který je možné k identifikaci použít.

8.2.1 PS regulátor s beznázovým přepínáním a dynamickým omezením sumační složky

Regulátor vychází ze struktury klasického PSD je modifikován vynecháním D složky. Absence diferenční složky je výhodná u silně zašuměných signálů, jako je v tomto konkrétním případě. Akční zásah je omezen na technologicky přijatelnou mez, z důvodu nebezpečí proudového přetížení motoru. Při překročení meze by došlo k sepnutí ochrany a servozesilovač by signalizoval poruchu, z tohoto důvodu je nezbytně nutné použít omezení akčního zásahu. Dalším požadavkem na regulátor je možnost přepínat mezi vnitřním regulátorem implementovaným v servozesilovači a vlastním regulačním algoritmem běžícím na PLC. Pro dynamické omezení sumační složky a použití přepínání mezi jednotlivými regulátory je tato vazba doplněna o konstantu sledování, jež do jisté míry ovlivňuje chování soustavy. Perioda vzorkování je závislá na periodě cyklu programu, s touto periodou jsou čteny aktuální hodnoty a zapisovány vypočítané akční zásahy. Pro dosažení nejlepších výsledků by bylo zapotřebí vzorkovat se stejnou, periodou s jakou pracuje vnitřní regulátor v servozesilovači, tj. $400\mu\text{s}$. Komunikace za použití Ethernetu s úpravou Ethernet POWERLINK je schopna poskytnout dostatečně rychlou odezvu nezbytnou pro regulaci servopohonů, i přes to je nutné nastavit komunikaci na násobek základní periody $400\mu\text{s}$, s touto periodou pracuje NC Manager, který se stará o čtení a zápis dat a se stejnou periodou je prováděn výpočet programu. V konkrétním případě je perioda vzorkování T_{vz} nastavena na $4000\mu\text{s}$. Akční zásah je omezen na hodnotu ± 5000 .



Obrázek 8.3 Struktura PS regulátoru

Základní rovnice PS regulátoru:

$$u(k) = K \cdot \left(e(k) + \frac{T_{VZ}}{T_i} \sum_{i=1}^k e(k) \right) \quad (8.2.1)$$

$$e(k) = \text{žadana} - \text{aktuální} \quad (8.2.2)$$

Odchylka vzniká jako rozdíl žádané polohy a aktuální polohy poskytované resolverem v příslušném výpočetním kroku.

8.2.2 Nastavení parametrů regulátoru

Pro prvotní určení konstant regulátoru byla zvolena metoda, která by v reálném procesu pravděpodobně nebyla realizovatelná s ohledem na připojené technologie, metoda Ziegler-Nichols. Vnitřní polohový regulátor byl nahrazen proporcionálním regulátor, běžícím na PLC. Zesílení bylo postupně zvyšováno až na mez stability, kde se objevily netlumené kmity.

Byly odečteny tyto hodnoty:

- Kritické zesílení $K_{krit} = 145$
- Perioda vlastních kmitů $T = 0,044$ [s]

Z těchto hodnot byly vypočteny konstanty regulátoru podle doporučeného nastavení:

$$\begin{array}{ll} P = 0,6K_{krit} & \\ \text{PID: } T_I = 0,5T_{Krit} & \text{PI: } K = 0,45K_{krit} \\ T_D = 0,125T_{Krit} & T_I = 0,83T_{krit} \end{array}$$

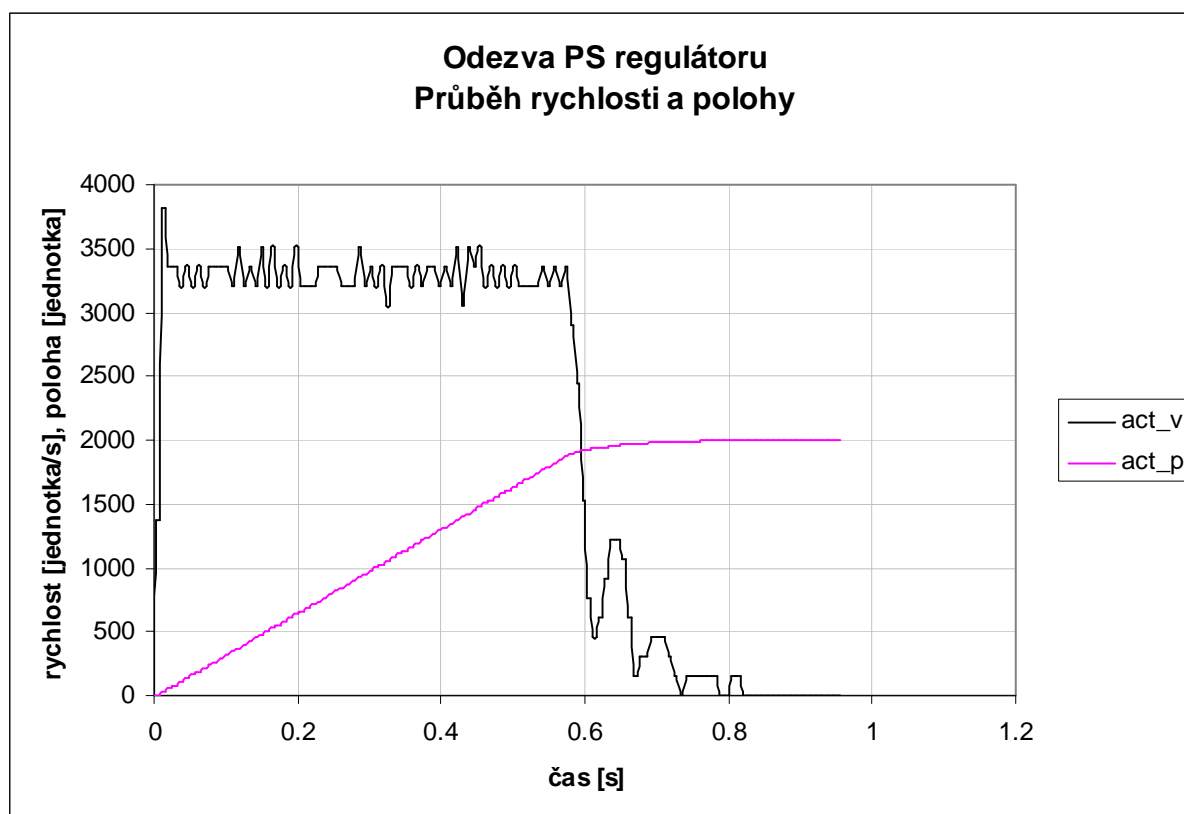
	pro PID	pro PI
K	87	65,25
T_I [s]	0,022	0,03652
T_D [s]	0,0055	-

Tabulka 8.5 Parametry regulátoru podle metody Ziegler-Nichols

Výchozí parametry poskytnuté měřením podle metody Ziegler-Nichols, byly dále modifikovány, aby bylo dosaženo přechodného děje bez překmitu a aby se optimalizovala doba trvání přechodného děje.

Proporcionální složka	K_p	85
Sumační složka	Tip [s]	0,08
Konstanta sledování	T_t [s]	1
Perioda vzorkování	T_{vz} [ms]	4
Omezení sumační složky		± 5000

Tabulka 8.6 Parametry PS regulátoru

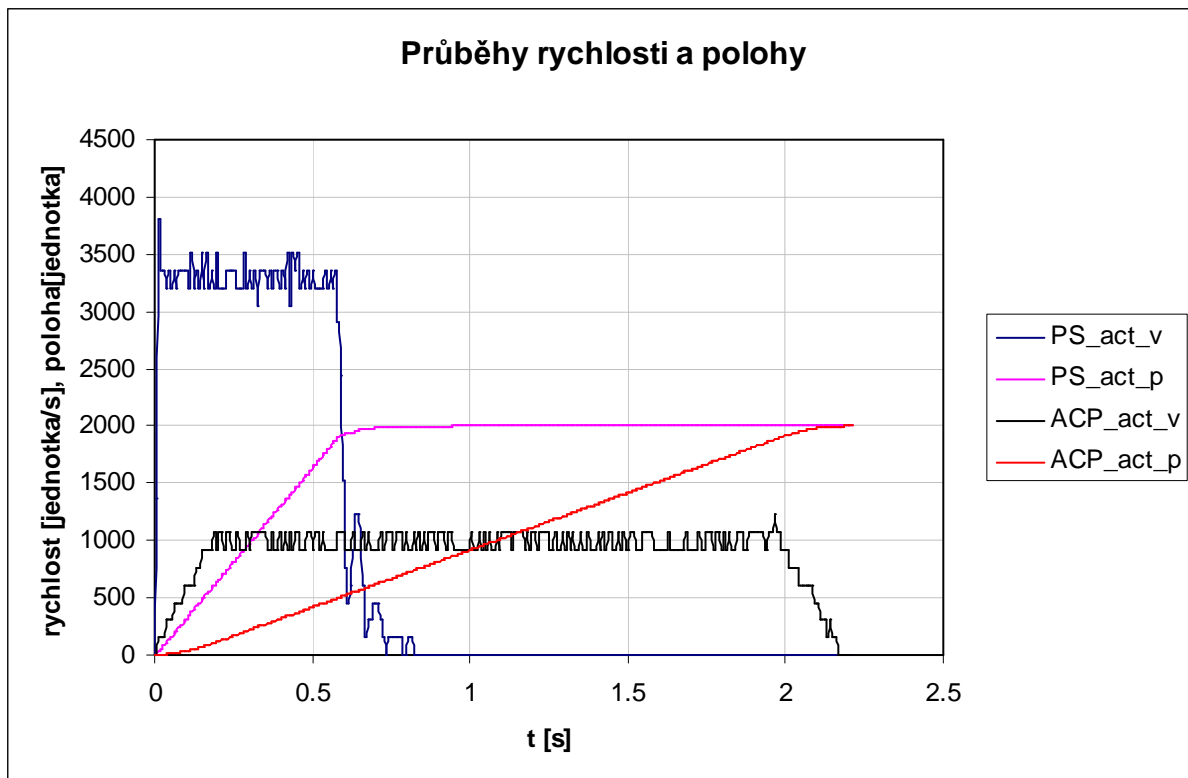


act_v – aktuální rychlost, act_p – aktuální poloha

Obrázek 8.4 Odezva PS regulátoru polohy a vnitřního rychlostního reg.

8.3 SROVNÁNÍ ODEZVY REGULÁTORŮ

Pro obě regulační smyčky jsou totožné podřízené regulátory proudu a rychlosti, které jsou nastaveny se stejnými parametry. Náhrada vnitřního regulátoru typu PI, regulátorem formálně stejného typu realizovaného na PLC není zcela ekvivalentní. Není známa přesná realizace diskrétního ekvivalentu spojitého PI regulátoru.



PS_act_v : průběh rychlosti při použití vlastního PS regulátoru

PS_act_p : průběh polohy při použití vlastního PS regulátoru

ACP_act_v : průběh rychlosti při použití vnitřního regulátoru

ACP_act_p : průběh polohy při použití vnitřního regulátoru

Obrázek 8.5 Srovnání průběhů obou regulátorů

Pro použití polohového regulátoru implementovaného v servozesilovači platí:

Z časového průběhu aktuálních hodnot je vidět značná závislost doby trvání přechodného děje v závislosti na nastavených parametrech pohybu při realizace použitím vnitřního regulátoru, zde byla omezena maximální rychlost na hodnotu 1000 jednotek/sekundu. Zvyšováním tohoto omezení by bylo možné celý přechodný děj zkrátit, maximální hodnota akcelerace, decelerace je také omezena na hodnotu 500 jednotek/s². Samotné nastavení regulátoru ovlivňuje chování soustavy z hlediska potlačení kmitů, nebo vyregulování odchylky polohy. Průběh regulačního děje z hlediska strmosti je ovlivnitelný převážně podle nastavení parametrů pohybu a nastavení regulátor nemá pro toto chování významný vliv.

Je-li použit regulátor implementovaný na řídicím PLC je vidět, že akční zásah je omezen jen vlastním omezením akčního zásahu regulátoru, dochází tedy k jeho prudkému nárůstu a

maximální otáčky jsou dosaženy téměř skokově. Při poklesu rychlosti otáček před dosažením žádané polohy dochází k mírnému prodloužení času potřebnému k jejich dosažení, zvětšením zesílení by bylo možné tuto část průběhu eliminovat, docházelo by však k překmitu žádané hodnoty. Zvětšením integrační konstanty by bylo možné dosáhnout plynulejšího průběhu nárůstu a poklesu rychlosti, následkem by však bylo prodloužení přechodného děje, nastavení regulátoru bylo voleno jako kompromis mezi těmito dvěma požadavky.

9. ZÁVĚR

Servozesilovač ACOPOS od firmy B&R poskytuje ucelenou platformu pro vývoj aplikací pro přesné polohování. Výhodou je možnost zapojení do systému jiného výrobce, tak poskytnutí všech dalších nezbytných komponent pro tvorbu systému pro řízení. Primární uplatnění se nalézá v automatizaci strojů, kde převládá využití střídavých pohonů.

V úvodu této bakalářské práce je pojednáno o vnitřní struktuře servozesilovače spojující v sobě silovou část pro napájení synchronního motoru pulsně šířkově modulovaným signálem a řídicí část, jenž obsahuje výpočetní logiku s vlastním operačním systémem. Výpočetní logika realizuje výpočty tří kaskádně zapojených regulátorů. Informaci o poloze rotoru lze získat z různých typů čidel, díky použití zásuvných modulů, jenž realizují interface k připojení do servozesilovače. V demonstrační úloze je použit resolver, jenž poskytuje relativní informaci o poloze rotoru, princip snímání je popsán v kapitole dvě. Součástí servozesilovače je i výpočet matematického modelu teplotního zatížení vinutí motoru, z důvodu zabránění přílišnému tepelnému namáhání. Toto opatření zvyšuje životnost a chrání před destrukcí. Připojený motor je synchronní s permanentními magnety na rotoru a snímačem polohy resolverem implementovaným na hřídeli. K ovládání slouží PLC Power Panel s dotykovou obrazovkou spojené pomocí realtime sběrnice Ethernet Powerlink.

K vývoji aplikace slouží vývojové prostředí Automation studio. Tento sofistikovaný nástroj umožňuje vytvářet programy v řadě programovacích jazyků, konkrétně byl použit jazyk ANSII C. Pro tvorbu vizualizační aplikace slouží komponenta, která je součástí Automation studia a ve které byla vytvořena aplikace, pomocí níž je skrze dotykový panel možno ovládat motor.

V textu je popsána struktura NC koncept, zahrnující komponenty pro řízení pohonů. Servozesilovač má vlastní operační systém, pro komunikace mezi ním a uživatelským programem se používá operační systém NC manager, běžící na řídicím PLC. Obousměrná komunikace probíhá na bázi příkazů pro vykonání akce pro servozesilovač. Stav vykonané akce je vrácen zpět do PLC. K řízení se používají funkce knihovny PLCopen. Tato knihovna poskytuje základní příkazy pro vytvoření datové struktury, jenž reprezentuje řízenou osu a funkce pro její ovládání. Tato knihovna je rozšířena o funkce vyvinuté firmou B&R, tyto funkce se uplatňují při požadavku na změnu vnitřních parametrů servozesilovače, které by byly standardními funkcemi nepřístupné. Každý parametr je identifikován unikátním číslem, pomocí kterého se uskutečňuje přístup pro čtení a podle druhu parametru i zápis. Díky těmto

funkcím je možné vypnutí polohového, nebo i rychlostního regulátoru a nahrazení regulátorem vlastním.

Náhrada polohového regulátoru, implementací vlastního regulačního algoritmu byla uskutečněna a je ověřena schopnost zařízení takto pracovat.

Pro dosažení optimálních průběhů regulovaných veličin, v tomto případě rychlosti a polohy je zapotřebí správné nastavení regulátorů. Servozesilovač obsahuje tři kaskádně zapojené regulátory typu PI. Parametry regulátoru proudu se nastavují automaticky podle provozních podmínek. Jemu nadřazený je regulátor rychlosti, zde bylo nutno nastavit hodnotu zesílení a sumační složky. Další volitelnou součástí je nastavení mezní frekvence filtru aktuální měřené veličiny. Nastavení regulátoru bylo realizováno s požadavkem na omezení kmitů rychlosti. Třetím nadřazeným regulátorem je polohový regulátor, zde k parametrům proporcionální a sumační složky je možno nastavit zpoždění a dopřednou vazbu pro spojení žádané hodnoty přímo s akčním zásahem. Regulátor byl nastavován s požadavkem přechodného děje bez překmitu. Žádaná hodnota pro polohový regulátor vzniká v bloku generátor žádané hodnoty, jenž generuje průběh žádané hodnoty pro optimalizaci chování soustavy.

Součástí práce bylo i implementovat vlastní regulátor do PLC, který by nahrazoval polohový regulátor v servozesilovači. Díky použití rychlé realtime sběrnice a s využitím speciálních funkcí pro čtení a zápis do parametrů regulátorů se podařilo realizovat PS regulátor s dynamickým omezením sumační složky, nastavení probíhalo s požadavkem na zamezení překmitu při přechodném ději. Zpětnou vazbu realizuje resolver na hřídeli motoru.

Srovnání chování obou regulátorů není zcela ekvivalentní, není známa přesná diskrétní realizace spojitě popsaného PI regulátoru a vlastnosti co do rychlosti přechodného děje za použití vnitřních regulátorů jsou omezené hodnotami maximálního zrychlení a rychlosti, které je možné zadat. Nastavení regulátorů se projevovalo jen eliminací překmitu a zamezení kmitání. Při použití vlastního PS regulátoru je snaha o maximální akceleraci na začátku a maximální deceleraci na konci přechodného děje, která je omezena jen na hodnoty, které může motor poskytnout. Další ladění, nebo implementace jiného algoritmu by bylo závislé na vlastnostech a požadavcích připojené technologie. Program s možností odpojení a náhradou vnitřních regulátorů lze využít jako učební pomůcku pro výuku návrhu a ladění diskrétních regulátorů.

10. LITERATURA

- [1] *The Basics of Motion Control*. B&R Automation 2005
- [2] *The Basics of ASiM*. B&R Automation 2005
- [3] *ASiM Basics Function*. B&R Automation 2005
- [4] *ACOPOS ACP10 Software*. B&R Automation 2005
- [5] *ACOPOS Control Concept and Adjustment*. B&R Automation 2005
- [6] *The Basics of ASiV*. B&R Automation 2005
- [7] *Automation Help*. B&R SYSTEM 2005
- [8] SKALICKÝ, J. *Elektrické servopohony*. 2. vydání. VUT Brno 2001.
ISBN 80-214-1978-4
- [9] *8MS Three-phase Synchronous Motors User's Manual* Version: 1.0 (září 2004)
- [10] Pivoňka, P. *Číslicová řídicí technika* [elektronická skripta] 2003-11-1. Brno
- [11] Veleba, V. *Číslicová řídicí technika, Počítačová cvičení*. [elektronická skripta] 2005-10-1. Brno
- [12] *B&R Automation* [online] 2008
<http://www.br-automation.com>
- [13] *PLCopen for efficiency in automation* [online], 2008
<http://www.plcopen.org>

SEZNAM ZKRATEK A SYMBOLŮ

Zkratka/Symbol	Jednotka	Popis
act_p	Jednotka	Aktuální poloha
act_v	Jednotka.s ⁻¹	Aktuální
PS_act_v	Jednotka.s ⁻¹	Rychlosti při použití vlastního PS reg.
PS_act_p	Jednotka	Poloha při použití vlastního PS reg.
ACP_act_v	Jednotka.s ⁻¹	Rychlost při použití vnitřního reg.
ACP_act_p	Jednotka	Poloha při použití vnitřního reg.
Kp	-	Proporcionální složka PS reg.
Tip	s	Sumační složka PS reg.
Tt	s	Konstanta sledování PS reg.
Tvz	ms	Perioda vzorkování PS reg.
P	s ⁻¹	Proporcionální složka PI reg.
I	s	Integrační složka PI reg.
Tf	s	Mezní perioda filtru PI reg.

SEZNAM PŘÍLOH

Příloha č. 1 - Přiložené CD.....	- 47 -
Příloha č. 2 – Ilustrační obrázky zařízení.....	- 47 -
Příloha č. 3 - Výpis programu pro čtení a zápis do struktury regulátorů.....	- 49 -

Příloha č. 1 - Příložené CD

Součástí bakalářské práce je příložené CD obsahující tyto adresáře:

Řídicí SW

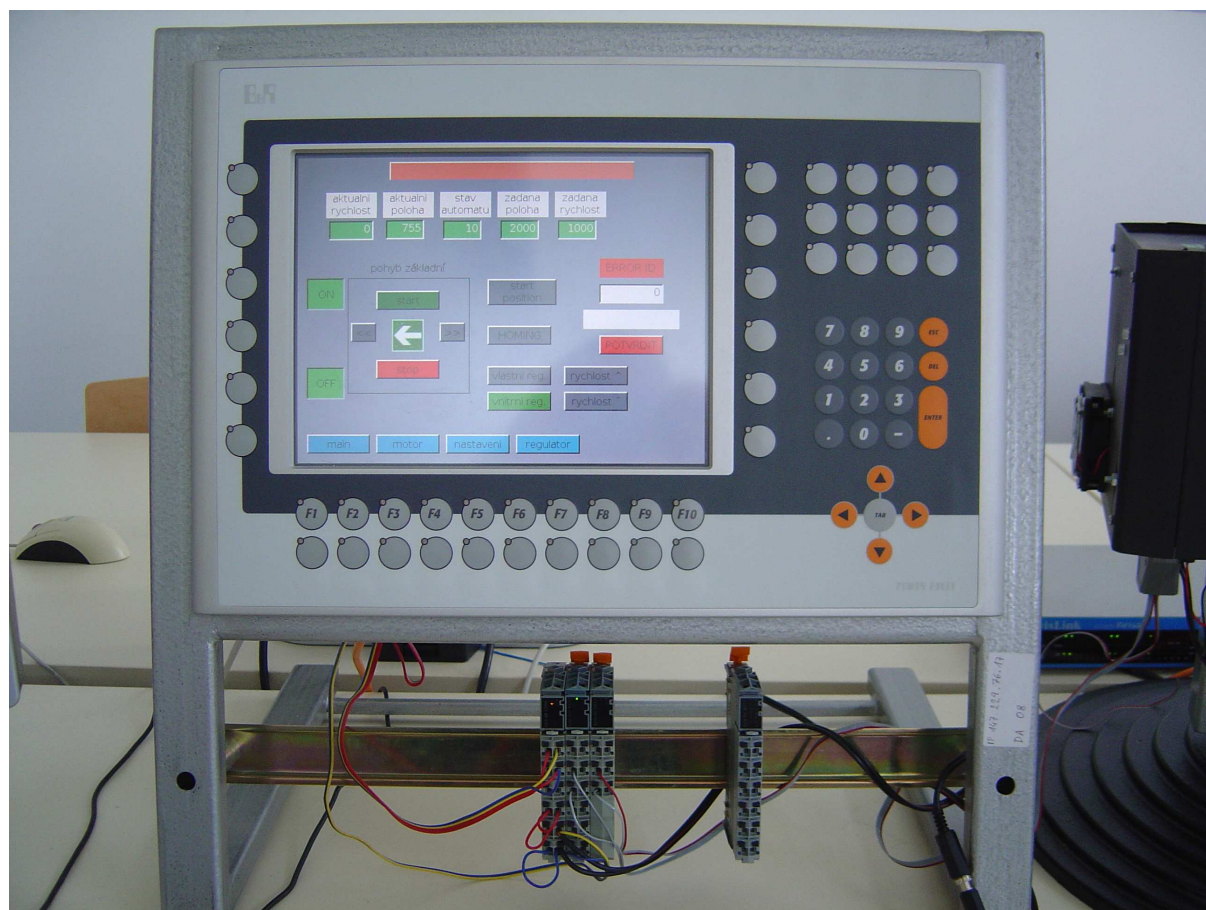
Řídicí program pro Power Panel, obsahuje program pro řízení motoru, vizualizaci a další nezbytné softwarové součásti pro řízení a definici zařízení.

Text

Text bakalářské práce v elektronické podobě.

Příloha č. 2 – Ilustrační obrázky zařízení

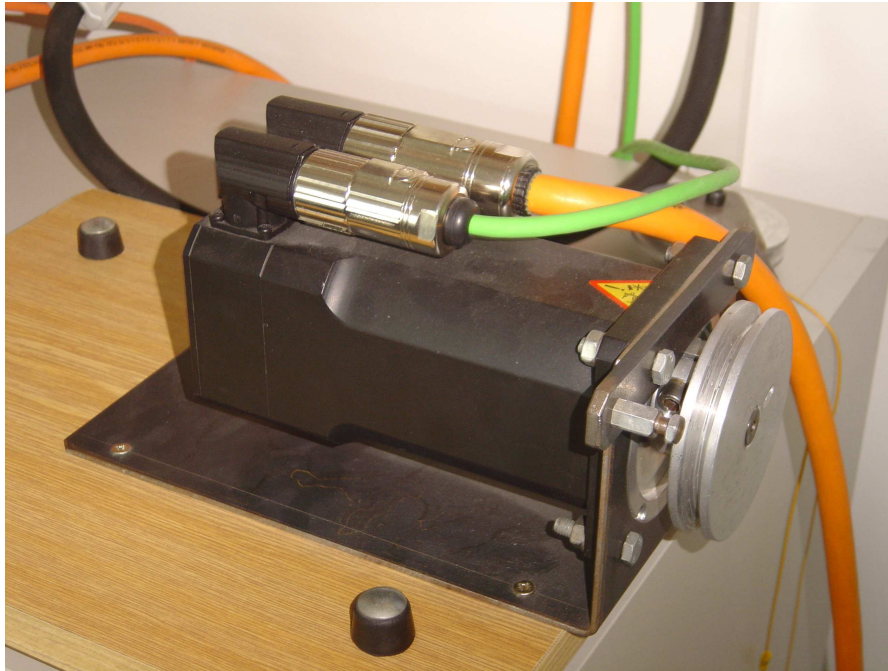
Power Panel PP282



Servozesilovač ACOPOS 8V1022.00-2



Synchronní motor 8MSA3L.R0-30 od firmy B&R



Příloha č. 3 - Výpis programu pro čtení a zápis do struktury regulátorů

```
#define act_pos                111
#define pos_en                 231
#define set_speed              329

_LOCAL      MC_BR_InitCyclicRead_typ      MC_BR_InitCyclicRead_actP;
_LOCAL      MC_BR_InitCyclicWrite_typ     MC_BR_InitCyclicWrite_setS;
_GLOBAL    UDINT      Axis1Obj;
_LOCAL     DINT        act_p;
_LOCAL     UINT        pos_off;
_LOCAL     REAL        set_s;

// cyklická část programu
// čtení aktuální hodnoty polohy
MC_BR_InitCyclicRead_actP.Axis = Axis1Obj;
MC_BR_InitCyclicRead_actP.ParID = act_pos;
MC_BR_InitCyclicRead_actP.DataType = ncPAR_TYP_DINT;
MC_BR_InitCyclicRead_actP.DataAddress = (UDINT)&act_p;
MC_BR_InitCyclicRead(&MC_BR_InitCyclicRead_actP);
if (MC_BR_InitCyclicRead_actP.Done == 1)
{
    MC_BR_InitCyclicRead_actP.Execute = 0;
    chyba_cteni= 0;
}
```

```

if (MC_BR_InitCyclicRead_actP.Error == 1)
{
MC_BR_InitCyclicRead_actP.Execute = 0;
chyba_cteni =1;
}
// vypnutí polohového regulátoru
MC_BR_WriteParID_posOFF.Axis= Axis1Obj;
MC_BR_WriteParID_posOFF.ParID = pos_en;
MC_BR_WriteParID_posOFF.DataType = ncPAR_TYP_UINT;
MC_BR_WriteParID_posOFF.DataAddress = (UDINT)&pos_off;
MC_BR_WriteParID(&MC_BR_WriteParID_posOFF);
// implementace regulačního algoritmu
:
:
:
// zápis akčního zásahu od polohového regulátoru
MC_BR_InitCyclicWrite_setS.Axis = Axis1Obj;
MC_BR_InitCyclicWrite_setS.ParID = set_speed;
MC_BR_InitCyclicWrite_setS.DataAddress = (UDINT)&set_s;
MC_BR_InitCyclicWrite_setS.DataType = ncPAR_TYP_REAL;
MC_BR_InitCyclicWrite(&MC_BR_InitCyclicWrite_setS);

```