

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Studijní program: M 2612 – Elektrotechnika a informatika

Studijní obor: 3902T005 – Automatické řízení a inženýrská informatika

Analýza řízení otáček elektrických pohonů firmy Siemens ovládaných pomocí průmyslové komunikační sběrnice PROFIBUS

Analysis of Siemens electric drives turn control operated by industrial communication bus the PROFIBUS

Diplomová práce

Autor: Pavel Bureš

Vedoucí diplomové práce: Doc. Ing. Pavel Rydlo, Ph.D.

Konzultanti: Ing. Martin Diblík

Ing. Petr Přívratský

Zadání DP

Prohlášení

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

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé DP a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom(a) toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

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

Datum

Podpis

Poděkování

Při řešení diplomové práce bych rád poděkoval za pomoc vedoucímu diplomové práce Ing. Pavlu Rydlovi a konzultantům Ing. Petru Přívratskému a Ing. Martinu Diblíkovi, že mi umožnili získat cenné odborné znalosti formou konzultací a technické dokumentace.

Anotace

Diplomová práce se zabývá analýzou řízení otáček elektrických pohonů firmy Siemens pomocí průmyslové sběrnice PROFIBUS.

Cílem práce je provést časovou analýzu generování otáček PLC automatem v závislosti na komunikačních parametrech a počtu řízených motorů. Touto analýzou získám vztah mezi parametry sběrnice, počtem motorů a kvality řízení, kterým bude možné určit za jakých podmínek a jak je tento způsob řízení vhodný pro řešení aplikací řízení více pohonů.

Pro ovládací automat PLC byl pro potřeby měření vytvořen řídicí program do PC s použitím technologie OPC která umožňuje vzdálený přístup pomocí internetu a podpůrné knihovní funkce pro PLC automat k univerzálnímu použití, pro vzdálené řízení pohonů pomocí PLC a následné zpětné zobrazení naměřených hodnot tak aby bylo možné měřit bez fyzické přítomnosti u zařízení.

Abstract

Analysis of Siemens electric drives turn control operated by the PROFIBUS protocol in view of various demands of configuration. Simulation obtained results by computer model. Software support for remote measurement by OPC technology created function for computer and PLC.

The aim of diploma work is to make time analysis of generating turns by PLC in view of various demands communication parameter and number of regulated motors. Create relation between parameters of communications, number of motors and quality precision of regulation to define how much regulation is convenient to solve any problem. Create main program for PC to control PLC with OPC technology, that make possible remote access via internet and create library functions for PLC to remote control electric motors by PLC and reverse show measured values without requested physical presence at the device.

Obsah

Úvod.....	9
1. Rozbor pracoviště.....	10
1.1. Popis sběrnice PROFIBUS	11
1.1.1 Komunikace PROFIBUS-DP Master – Slave	13
1.2. Popis PLC S7-300.....	15
1.3. Popis frekvenčních měničů	16
1.4. Technologie OPC.....	17
2. Programové vybavení.....	19
2.1. Program a nastavení PLC	19
2.1.1 Nastavení v HW Config a NetPro.....	20
2.1.2 Generátor průběhu	21
2.1.3 Vytváření DB bloků.....	22
2.1.4 Kopírování DB bloků.....	22
2.1.5 Indexování ukládání a čtení hodnot v tabulce	23
2.1.6 Kopírování dočasných hodnot s MMC kartou.....	23
2.1.7 Ostatní podpůrné bloky.....	23
2.1.8 Hlavní program	24
2.2. Program a konfigurace OPC	26
2.2.1 Vývoj programu v Control Web	27
3. Analýza dynamiky řízení.....	30
3.1. Analýza komunikace.....	30
3.1.1 Teoretický rozbor analýzy komunikace.....	30
3.1.2 Změření a analýza dynamiky komunikace	32
3.1.3 Vytvoření simulačního modelu.....	38
3.2. Posouzení kvality řízení.....	41
3.2.1 Výpočet kritéria Kvality	43
3.2.2 Obecné určení kvality	44
3.3. Ověření modelu a vztahu kvality	46
4. Porovnání různých metod řízení.....	48
5. Závěr.....	50
Seznam použité literatury.....	51
Seznam obrázků:	52

Seznam Tabulek	53
Příloha A – Seznam a popis napsaných funkcí	54
Příloha B –Popis řídicí aplikace	78

Seznam termínů, zkratk a symbolů

PLC	Programmable Logic Controller	Programovatelný logický automat
OPC	OLE for process control	technologie sdílení dat
MPI	Multi point interface	rozhraní vyvinuté firmou Siemens
PPO	Parameter process data object	definice objektu procesních dat
MMC	Micr memory card	elektronické paměťové médium
STL	Statement list	programovací jazyk symbolických adres
FM	frevenční měnič	elektronické zařízení sloužící k ovládání otáček elektrických pohonů
Master		Nadřízené zařízení při komunikaci
Slave		Podřízené zařízení při komunikaci
Baud		Jednotka míry pro modulační rychlost signálu, udává množství informace za vteřinu, pro binární rychlosti roven přenosové rychlosti(1 Baud = 1 b/s)
amplituda generovaného průběhu otáček ku jmenovitým otáčkám	A	[%]
střední hodnota otáček ku jmenovitým otáčkám	O _{Str}	[%]
perioda generovaného průběhu	T	[s]
čas cyklu komunikace	T _{Cykl}	[s]
přenosová rychlost	C	[Baud]
počet řízených měničů	PM	[-]
čas komunikace obsluhy jednoho FM	T _{FM}	[s]
počet bytů směřujících komunikaci	N _K	[-]
počet bitů na přenos jednoho bytu	N _B	[b/B]
počet datových bytů pro FM	N _R	[-]
čas cyklu programu PLC	T _P	[s]
plocha mezi křivkami generovaného a skutečného průběhu	S _R	[-]
plocha ohraničená generovaným průběhem a hodnotou středních otáček.	S _G	[-]
Kvalita průběhu	Kvalita	[%]

Úvod

Katedra elektrotechniky na fakultě mechatroniky Technické univerzity v Liberci se s postupným rozvojem průmyslových komunikačních technologií zabývá komunikační sběrnici PROFIBUS na využití k řízení otáček elektrických pohonů na řídicích systémech firmy Siemens pro zdokonalení a zjištění omezení řízení procesů.

PROFIBUS je poměrně rychlý, otevřený komunikační standart, umožňující spojení mezi řídicím systémem PLC a několika podřízenými jednotkami měničů. I přes vysokou komunikační rychlost zde vyvstávají problémy na komunikačním řetězci mezi automatem PLC a frekvenčními měniči s kvalitou odezvy na generovaný průběh otáček v závislosti na počtu měničů, nastavených komunikačních parametrech a doby reakce řídicího systému. Tyto důvody vedly k zadání této diplomové práce, jenž se po analýze snaží pro dané problémy určit mezní parametry sběrnice k dané kvalitě průběhu a zároveň uzavírá okruh prací, které byly zaměřené na analýzu podobných komunikačních technologií použitých k řízení otáček elektrických pohonů. Dále se práce zabývá řešením otázky vzdáleného měření a vizualizace pro aplikace řízení pohonů s využitím technologie OPC.

V začátku práce bylo vytvořeno pracoviště s PLC řady S7-300, frekvenčním měničem s pohonem a počítačem vybaveným prostředky k použití technologie OPC tak aby bylo možné vytvoření univerzálně použitelných funkcí pro řídicí PLC k řízení pohonů a spojení s technologií OPC pro kterou byl vytvořen program k nastavení řízení PLC, získání a vizualizaci naměřených dat. Následně byly analyzovány možnosti sběrnice PROFIBUS k ovládání měničů a provedena časová analýza v závislosti na nastavení komunikace a počtu ovládaných frekvenčních měničů. Z analýzy byl vytvořen počítačový model k možnosti získání omezení a výsledků simulací u hodnot parametrů, při kterých nebyla analýza provedena. Dosažené výsledky budou aplikovány na vyhodnocení kvality pomocí simulací na vytvořeném modelu.

1. Rozbor pracoviště

Po konzultaci o dostupném zařízení laboratoře, které mohu použít, jsem byl vybaven pro tvorbu funkcí a programů:

- automatem PLC firmy Siemens řady S7–300 vybaveným rozhraními pro sběrnice MPI a PROFIBUS
- frekvenčním měničem MicroMaster 440, s připojeným asynchronním motorem, vybaveným modulem s rozhraním pro sběrnici PROFIBUS
- osobním počítačem vybaveným komunikační kartou, s rozhraním umožňujícím komunikaci na sběrnici MPI nebo PROFIBUS nastavenou na PROFIBUS, a technologií OPC server, umožňující čtení a zápis dat v automatu PLC a práci s těmito daty přes síť internet pomocí programu ovládajícího technologii OPC.
- osobním počítačem vybaveným rozhraním sběrnice MPI a vývojovým prostředím SIMATIC STEP 7 pro konfiguraci a programování automatu PLC a vývojovou aplikaci ControlWeb, podporující technologii OPC klient, pro vytvoření programu k řízení a vizualizaci procesů automatu PLC.

Po dokončení vývoje řídicích a vizualizačních programů bude pro časovou analýzu s větším počtem řízených pohonů pracoviště doplněno postupně:

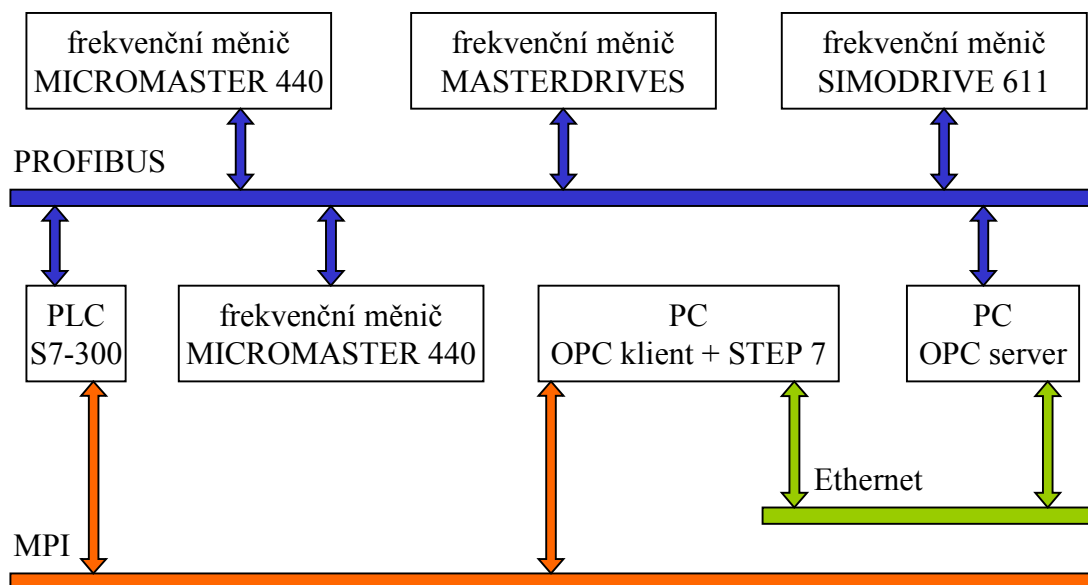
- frekvenčním měničem MicroMaster 440, s připojeným asynchronním motorem, vybaveným modulem s rozhraním pro sběrnici PROFIBUS

pro časovou analýzu s řízením 2 pohonů a pak následně doplněno pro časovou analýzu s řízením 4 pohonů:

- frekvenčním měničem SIMODRIVE 611 universal E, s připojeným servomotorem, s modulem s rozhraním pro sběrnici PROFIBUS
- frekvenčním měničem SIMOVERT MASTERDRIVES, s připojeným servomotorem, s rozhraním pro sběrnici PROFIBUS

Vzhledem k tomu, že na propojení sběrnicemi nebylo co řešit a bylo dané, následující obrázek (Obr.1-1) znázorňuje schématické propojení jednotlivých zařízení sběrnicemi použitých na pracovišti, první z nich je sběrnice MPI (Multi-Point Interface), jedná se o sběrnici vyvinutou firmou Siemens na základě rozhraní RS485 sloužící k programování a sledování automatů PLC a ovládacích panelů, propojení mezi nimi ke sdílení dat, přičemž umožňuje komunikaci až mezi 8 zaříze-

ními, v této práci je však použita pouze k programování a sledování PLC, proto není potřeba se na ní více zaměřit. Druhou použitou sběrnici je PROFIBUS, kterou se zabývá část této práce a bude rozebrána podrobněji dále



Obr. 1-1: Kompletní propojení zařízení sběrnici

1.1. Popis sběrnice PROFIBUS

Sběrnice PROFIBUS je univerzálním otevřeným komunikačním standardem pro výrobní a řídicí automatizaci s nastavitelnou přenosovou rychlostí 9,6 kBaud až 12 MBaud s maximální vzdáleností až 1 200 m, jejíž omezení v závislosti na přenosové rychlosti je v následující tabulce, a možností připojit až 32 podřízených stanic.

Tab. 1.1-1: Závislost maximální délky na přenosové rychlosti

Přenosová rychlost [kBaud]	Maximální délka vedení [m]
9,6	1200
19,2	1200
45,45	1200
93,75	1200
187,5	1000
500	400
1500	200
3000	100
6000	100
12000	100

Z daného výčtu přenosových frekvencí jsem pro analýzu zvolil, s ohledem na možné porovnání s metodami řízení v předchozích pracích a s praktickým využitím při vysokých rychlostech přenosu, 19,2 kBaud, 45,45 kBaud, 187,5 kBaud, 1,5 MBaud.

Standardem jsou definovány 3 základní profily komunikace podle systémové úrovně komunikujících zařízení, pro všechny je společné zajištění přenosu dat, interpretace přenášených dat je však už věcí uživatelského programu a nastavení zařízení aby byla možná zaměnitelnost výrobců:

- **PROFIBUS-PA** (Process Automation) je koncipován pro řízení pomalých procesů v nebezpečném prostředí s umožněním napájení zařízení přímo ze sběrnice, s ohledem na bezpečnost je omezen počet připojených zařízení a pevná přenosová rychlost.
- **PROFIBUS-FMS** (Fieldbus Message Specification) je určen především ke komunikaci mezi řídicími stanicemi pro sdílení dat, umožňující propojení stanic různých výrobců. Komunikace probíhá formou předávání pověření, které určuje jaká stanice může aktivně řídit komunikaci.
- **PROFIBUS-DP** (Decentralized Periphery) je určen ke komunikaci se vzdálenými periferními zařízeními, kde jsou kladeny nároky na velkou rychlost přenosu a složitost komunikace je nízká. Na sběrnici může být umístěno několik řídicích automatů (Master), které si předávají pověření k řízení periférií sobě přiděleným (Slave), většinou je na sběrnici pouze jeden Master a komunikace probíhá přístupem Master – Slave.

V tomto profilu je komunikace rozdělena do tří úrovní. Základní úroveň DP-V0 zprostředkovává základní funkce obsahující cyklickou komunikaci, diagnostické funkce stanic, modulů a komunikace. Druhá úroveň DP-V1 rozšiřuje o neperiodickou komunikaci umožňující nastavování a správu chybových zpráv. Neperiodická komunikace je prováděna paralelně s cyklickou komunikací ovšem s nižší prioritou. To je prováděno pomocí předávání povolení ke komunikaci, kdy cyklicky komunikující zařízení po ukončení cyklu předá povolení druhému zařízení které pak může ve zbylém čase omezeným cyklem komunikace adresovat jakékoliv podřízené zařízení. Třetí úroveň DP-V2 rozšiřuje komunikaci o směrování mezi podřízenými stanicemi bez nutnosti komunikovat přes stanicí nadřizenou. Dalším rozšířením je globální synchronizace podřízených a nadřizené stanice, tato funkce umožňuje velice přesné nastavování procesů s odchylkou menší než jedna mikrosekunda. Dalším

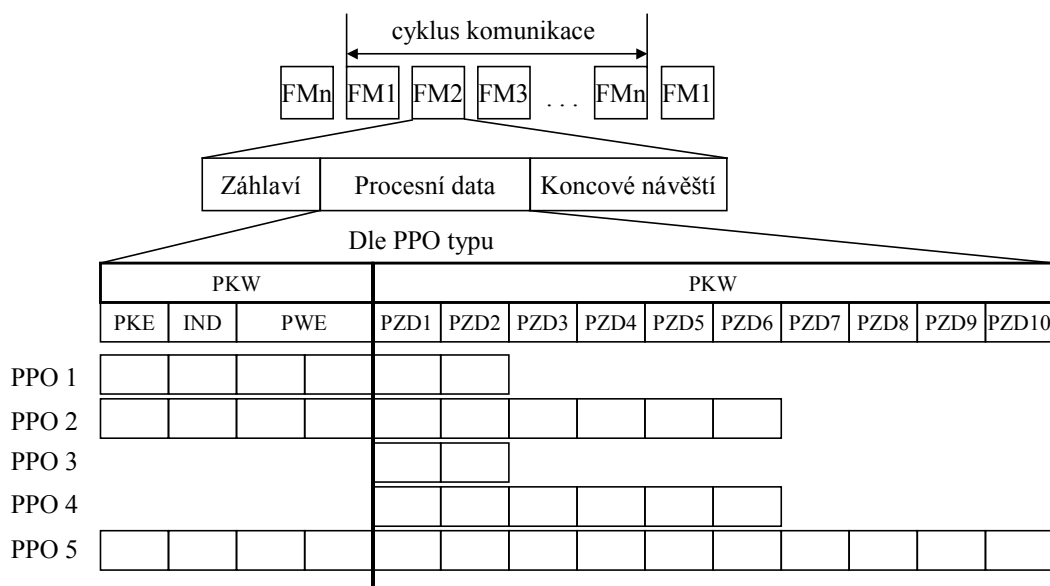
rozšířením je hodinové řízení, kdy nadřazená stanice vyšle podřízeným časovou značku pro synchronizaci na systémový čas s odchylkou menší než jedna milisekunda, to umožňuje přesné sledování událostí a získávání časových funkcí ze sběrnice s velkým počtem nadřazených stanic, což usnadní diagnózu chyb a správné plánování událostí.

Podle typu komunikace rozlišujeme tři typy zařízení. DPM1 (DP-Master Class 1) je zařízení, nejčastěji PLC automat, řídící zařízení typu Slave cyklickou výměnou dat. DPM2 (DP-Master Class 2) je zařízení, nejčastěji operační panel nebo OPC server, realizující konfigurační, diagnostické a monitorovací funkce se zařízením typu Master Class 1 pomocí acyklické komunikace, toto zařízení nemusí být ke sběrnici připojeno stále a chová se jako aktivní zařízení. Slave je podřízené zařízení zpracovávající posílané informace, vzhledem k sběrnici se chová pasivně a odpovídá pouze na přímý dotaz.

Pro komunikaci mezi zařízeními je použit profil PROFIBUS-DP vzhledem k jeho rychlosti a nenáročnosti komunikace a bude rozebrán podrobněji.

1.1.1 Komunikace PROFIBUS-DP Master – Slave

Komunikace probíhá na základě cyklického vysílání žádostí a odpovědí mezi zařízením Master (automat PLC) a jednotlivými Slavy (frekvenční měnič), které jsou adresovány pomocí kruhového seznamu (Obr.1.1.1-1), nastavení rychlosti komunikace se provádí v Masteru a podřízené stanice si při inicializaci sběrnice automaticky nastaví přenosovou rychlost přizpůsobením. Přenosová rychlost je stejná pro všechna zařízení na jedné smyčce PROFIBUSu.



Obr. 1.1.1-1: Protokol komunikace PROFIBUS-DP

Výměna Dat mezi Masterem a Slavem probíhá pomocí protokolu (Obr.1.1.1-1), začínajícím záhlavím, ve kterém jsou synchronizační data, komunikační parametry a typ struktury přenášených dat, následovaným procesními daty s danou strukturou, která může být pro každé zařízení Slave jiná a koncovým návěštím pro ukončení. Struktura procesních dat je dána typem PPO (Parameter process data object), který určuje kolik bude přenášeno wordů. Data jsou rozdělena do dvou částí, na oblast parametrickou (PKW), která slouží k nastavování nebo čtení parametrů měniče, obsahující 4 wordy s významem:

- **PKE** (WORD 1) určuje typ požadavku (čtení nebo zápis), délku slova v parametru PWE a číslo požadovaného parametru v měniči kmitočtu. V odpovědi informuje o průběhu vyřizování požadavku.
- **IND** (WORD 2) u indexovaných parametrů volí index vybíraného parametru.
- **PWE** (WORD 3-4) při požadavku zápisu určuje zapisovanou hodnotu, v odpovědi je pak totožná hodnota, při požadavku čtení je v odpovědi aktuální hodnota parametru. Parametr má délku slova 2 wordy, je-li požadavek pro přenos jednoho wordu, je přenášén v nižší části PWE

Druhou částí je procesní oblast, jejíž délka je volitelná podle typu PPO, a přenáší řídicí informace pro měnič a v odpovědi z měniče se posílají aktuální hodnoty stavů. Je definováno 5 typů PPO, z kterých lze zvolit podle potřeb komunikace, u měničů je však třeba dle dokumentace zjistit které typy podporuje, některé měniče s vyšším funkčním vybavením podporují vytvoření uživatelem definovaného typu PPO s volně definovanou délkou.

Interpretace přenášených dat mezi měniči a PLC se nastavuje pomocí parametrů ve frekvenčním měniči, možnosti nastavení jsou však závislé na druhu měniče, proto je vhodné prostudovat příslušnou dokumentaci. Z použitých frekvenčních měničů má MICROMASTER 440 nižší možnost nastavitelnosti než MASTERDRIVE a SIMODRIVE, volba typu PPO je omezena na typy 1-4 bez možnosti uživatelem definovat vlastní typ, příjem řídicích dat je pevně nastaven tak, že 1. word je řídicí slovo (CTW), ve kterém se nastavuje např. spouštění, ve 2. wordu je žádaná hodnota otáček. Vysílání aktuálních hodnot je nastavitelné s omezením na 1. word, ve kterém je stavové slovo (STW) obsahující stavové informace ovládacích členů v měniči. Vzhledem k možnostem nastavení měničů MASTERDRIVE a SIMODRIVE, budou nakonfigurovány stejně jako MICROMASTER, aby bylo možné provést analýzu s lineárně rostoucími nároky komunikace.

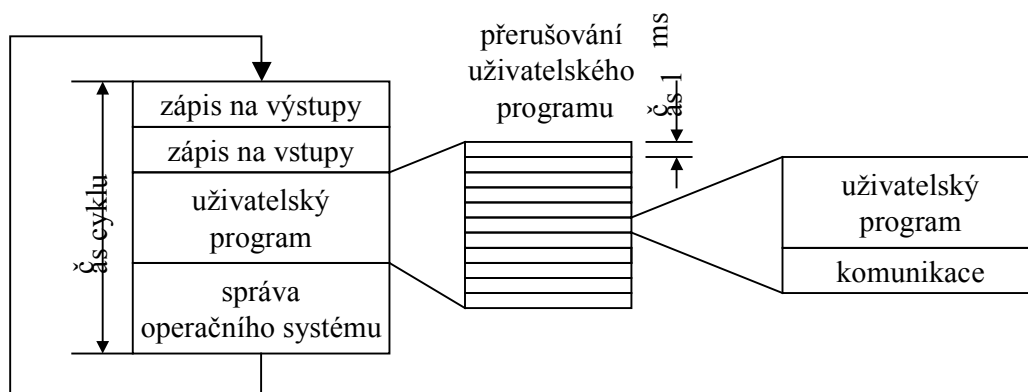
1.2. Popis PLC S7-300

Jedná se o řadu programovatelných logických automatů firmy Siemens pro řízení automatizace. Automaty se od sebe odlišují jak rychlostí matematických výpočtů, tak vybavením hardwaru. Pokud by některé vybavení nebylo dostatečné, je možné přidat rozšiřující moduly dle potřeby. Vzhledem k možné paměťové náročnosti při získávání časové analýzy uvádím výpis omezení použitého automatu CPU 313C-2 DP s tímto použitým vybavením:

- integrovaná RAM 32 KB
- slot Micro Memory Card (MMC) s pamětí 64KB

Dle mého názoru je zde vhodné se zmínit o časování programového cyklu, ve kterém probíhá komunikace sběrnice PROFIBUS sdílením času procesoru mezi vykonáváním uživatelského programu a správou komunikace. Každý cyklus programu začíná zápisem obrazů výstupů na rozhraní následovaný čtením vstupů do jejich obrazů, poté je spuštěn uživatelský program, po kterém je spuštěna správa systému k vykonání systémových funkcí. Uživatelský program je přerušován v úsecích 1 ms, tyto úseky jsou dále rozděleny na běh programu a komunikaci, poměr dělení úseků lze nastavit v hardwarové konfiguraci v rozmezí 5-50 % pro komunikaci. Z toho vyplývá, že pokud bude komunikace zatížena více frekvenčními měniči při stejném poměru rozdělení programu a komunikace, doba cyklu programu PLC bude zvýšena jen o nutný přenos dat do adresového prostoru pro

zařízení(Obr.1.2-1), ale doba k obsluze komunikace s měniči může být rozložena na několik programových cyklů PLC.



Obr. 1.2-1: časování cyklu PLC

Čas cyklů programu PLC je závislý na větvení programu, každý cyklus proto může mít jinou délku, z tohoto důvodu bude potřeba optimalizovat program tak, aby bylo oddělené případné volání složitějších funkcí, které přímo nesouvisí s částí programu na realizaci časové analýzy.

1.3. Popis frekvenčních měničů

Použité frekvenční měniče (FM) jsou výrobkem firmy Siemens a mají standardizované nastavování parametrů a funkcí. Popis tohoto nastavování a význam parametrů by bylo zacházení do podrobností, proto se zaměřím především na rozdíly mezi jednotlivými typy a hlavním nastavením pro správnou komunikaci po sběrnici PROFIBUS.

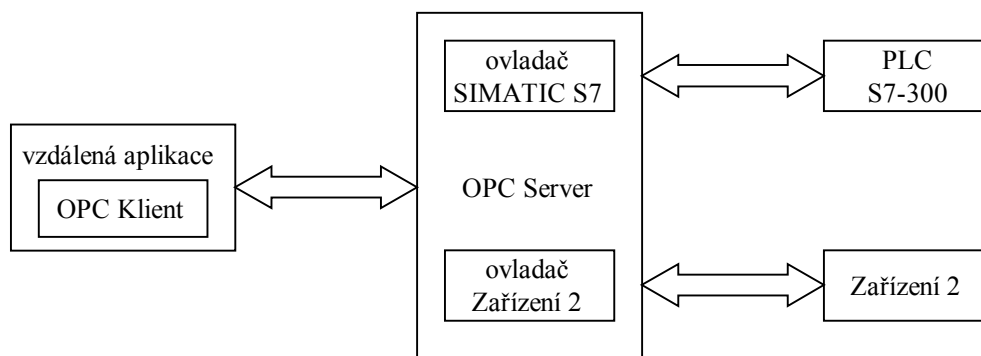
Všechny měniče jsou vybaveny zobrazovacím panel a tlačítky k zobrazení a nastavení jednotlivých parametrů, FM MASTERDRIVE a SIMODRIVE jsou navíc vybaveny rozhraním pro nastavení a vizualizaci pomocí obslužného softwaru na PC. Pro správnou komunikaci je třeba nastavit adresu, která musí být unikátní, pro směrování řízení na sběrnici PROFIBUS a nastavit správný zdroj žádaných hodnot. FM MICROMASTER je určen pro nižší náročnost řízení a jeho nastavitelnost je menší než u ostatních, proto bude nastavení řízení a komunikace přizpůsobeno tomuto typu FM. MASTERDRIVE a SIMODRIVE mají šířku slova žádané hodnoty nastavitelnou na 16 nebo 32 bitů, MICROMASTER má šířku slova nastavenou pevně na 16 bitů, proto se za cenu snížení jemnosti řízení omezím pouze na 16bitovou šířku slova, aby nebylo nutné zatěžovat PLC přepočty hodnot.

Měniče mají velkou možnost nastavení řízení průběhu otáček na motoru podle technologie v které jsou použity, pro tuto práci je však jejich využití minimální protože měřené motory jsou bez zatížení a jedná se především o analýzu dynamiky řízení. Přesto tu jsou parametry ovlivňující kvalitu analýzy nastavující dobu náběhu a doběhu motoru, které nastavují maximální směrnici náběhu, resp. doběhu motoru. Parametry jsou definovány časem, za který může motor změnit své otáčky z nulové hodnoty na hodnotu jmenovitých otáček. Pro analýzu byli tyto parametry nastavené na nulovou hodnotu a nebyli měněny mezi různými měřeními vzhledem k tomu že jejich nastavení by při vysokých přenosových rychlostech, kdy je spíše přebytek hodnot, mohlo zhoršit kvalitu měřených průběhů.

1.4. Technologie OPC

Jedná se otevřeně standardizované komunikační rozhraní mezi prvky průmyslové automatizace, jako jsou automaty PLC, a počítačovými řídicími systémy, využívající technologii OLE jako způsobu přenosu a sdílení informací mezi aplikacemi. Tato technologie není vázána na technické a programové vybavení jednotlivých výrobců průmyslové automatizace, které může být chráněnou uzavřenou technologií dané firmy, na které pak je zda vytvoří podporu pro OPC.

Technologie OPC využívá komunikace typu Klient-Server, kdy na počítači připojeného k systému automatizace pomocí příslušného vybavení, v našem případě pomocí rozšiřující karty s rozhraním PROFIBUS, je provozován OPC Server s podporou daného zařízení, který na žádosti vzdálené aplikace s podporou OPC Klient zprostředkovává sdílení dat mezi řídicím systémem automatizace a vzdálenou klientskou aplikací(Obr.1.3-1).



Obr. 1.3-1: Komunikace technologie OPC

Komunikace mezi OPC a PLC probíhá acyklicky formou žádostí od OPC a odpovědí od PLC, nelze tedy říci, že v každém cyklu bude provedena tato

komunikace. Z takto probíhající komunikace není možné sledovat rychlé změny dat, proto bude nutné, s přihlédnutím k ostatním omezením, vytvořit do programu PLC oblast dat a příslušných funkcí k záznamu dat časové analýzy, která pak bude postupně a nezávisle na měření přesunuta do vzdálené aplikace pomocí OPC.

2. Programové vybavení

Pro vývoj programů byla použita vývojová prostředí SIMATIC STEP 7 firmy Siemens a Control Web firmy Moravské přístroje. SIMATIC STEP 7 je použit pro vývoj programu do PLC, umožňující nastavení konfigurace připojených zařízení, tvorbu programu v PLC a sledování hodnot proměnných při běhu PLC pro možnosti ladění. Control Web je objektově orientované prostředí použité pro vývoj aplikace k řízení PLC a následné vizualizaci pomocí OPC technologie.

V této části bych rád především uvedl principy a důvody k vytvořeným algoritmům programů, se správnou funkcí aplikací však také souvisí konfigurace automatu PLC, proto bych rád uvedl, dle mého úsudku, hrubé rysy základů konfigurace.

2.1. Program a nastavení PLC

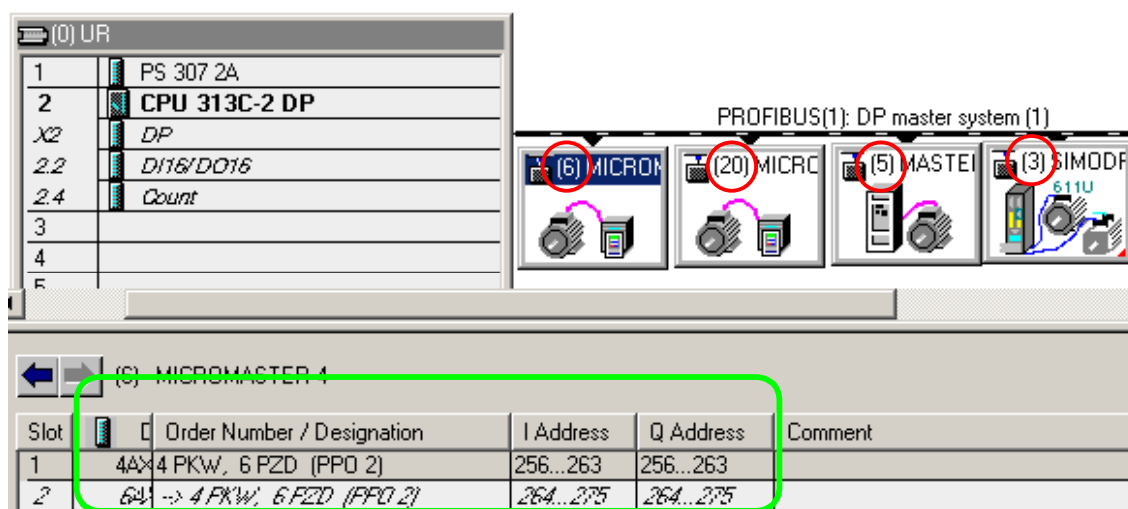
Programování automatů PLC lze provádět pomocí několika programovacích metod, vyšším programovacím jazykem obsahujícím pojmy obdobné z jazyka Pascal a ANSI C, grafickým zobrazením komplexních funkcí a jejich propojením, nebo pomocí textového jazyka se strukturou podobnou jazyku symbolických adres (STL). Programování probíhá metodou vytváření objektů s daty (DB) definující datovou strukturu, funkčních bloků (FB, FC) obsahující algoritmus funkce a organizačních bloků (OB) obsahující hlavní program a obslužné procedury. Vzhledem k osobním zkušenostem a preferencím jsem se rozhodl programovat pomocí textového jazyka STL v kterém lze dělat optimalizace na nejnižší úrovni.

Nastavení konfigurace PLC se provádí programy HW Config a NetPro. Konfigurace se realizuje vkládáním jednotlivých součástí zařízení v podobě grafických symbolů z databáze dodané s programem. Pokud je dané zařízení vyrobeno nebo inovováno později, než verze databáze, je potřeba tuto databázi doplnit informacemi o tomto zařízení instalací servisního balíku nebo standardizovaným souborem informací od výrobce.

2.1.1 Nastavení v HW Config a NetPro

V programu HW Config se nastavuje konfigurace daného PLC (typ, připojené moduly...) a k němu připojená zařízení (Obr.2.1.1-1). Pro správné nastavení a běh programu je především důležité:

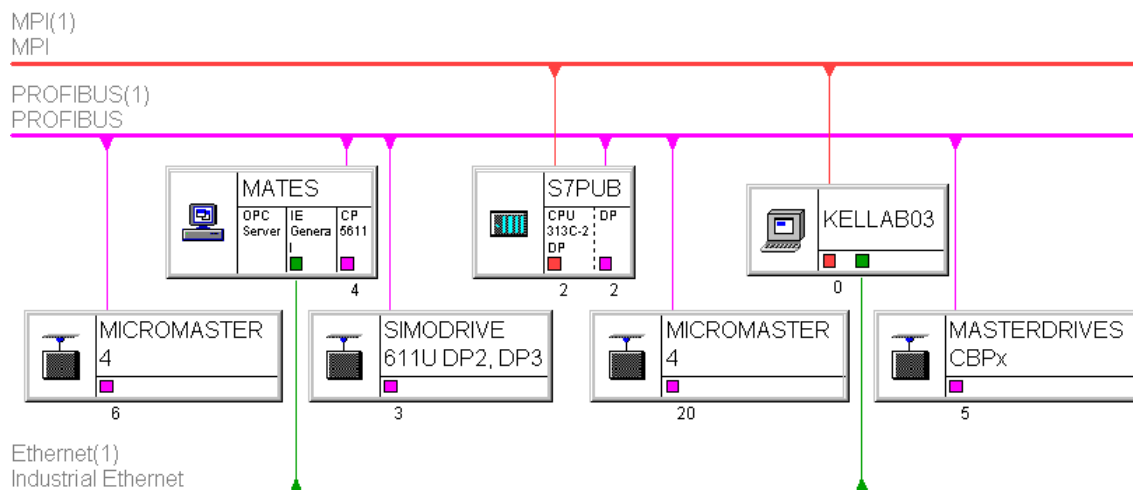
- nastavení adresy DP zařízení, která musí být shodná s nastavením adresy v DP zařízení a na připojené sběrnici PROFIBUS unikátní. Tato adresa je zobrazena u daného zařízení a na obrázku zvýrazněna červeným kroužkem
- adresový prostor v PLC pro komunikaci se zařízením, implicitně nastaven programem, ale měnitelný. Tyto adresy jsou označeny zeleným obdélníkem.



Obr. 2.1.1-1: konfigurace v programu HW Config

Program NetPro je nadstavbou programu HW Config a slouží k nastavení propojení sběrnic mezi zařízeními se stejnou úrovní řízení (Obr.2.1.1-2), nastavení parametrů sběrnic a v tomto případě i k nastavení adresových prostorů, které jsou pak sdílené pomocí OPC serveru.

Mimo tohoto nastavování umožňuje úplné propojení sběrnic automatickou změnu rozhraní pro nahrání konfigurace do daného zařízení, pro ilustraci je to volba rozhraní pro nahrání konfigurace do PC MATES nebo PLC S7PUB z PC KELLAB03, na kterém je toto vývojové prostředí spuštěno.



Obr. 2.1.1-2: Propojení sběrnic mezi stanicemi v prog. NetPro

2.1.2 Generátor průběhu

Pro generování signálu jsem zvolil sinusový průběh vzhledem k tomu že každý periodický signál lze rozložit pomocí fourierova rozvoje na součet sinusových průběhů s n-harmonickými frekvencemi. Tímto pak bude možné určit omezení frekvencí, kterými bude při dané kvalitě ještě možno řídit. Způsob časování generovaného signálu mohl být proveden:

- přerušením v cyklicky volaném přerušovacím organizačním bloku s nastavitelnou periodou přerušení 1–60 000 ms.
- čítačem a časovačem při nejmenším kroku 10 ms s rozsahem 10–9 990 ms.
- přičítáním času předchozího cyklu programu k aktuální hodnotě času, udávající uběhlý čas od spuštění generátoru.

Vzhledem k potřebám časové analýzy, kdy bude potřeba co nejmenší časový krok při komunikaci řízení, záznamu hodnot a generování signálu a zatím neznalosti rozsahu programu v PLC, jsem se rozhodl pro generování průběhu metodou přičítání času předchozího cyklu, kdy je v každém cyklu provedeno generování nové hodnoty průběhu, přečtení a záznam aktuální hodnoty otáček, kdežto pomocí předchozích dvou metod, s pevně nastaveným krokem, může dojít k vynechání výpočtu v době kdy by mohla být aktualizována hodnota průběhu a poslána ke komunikaci (krok generátoru větší než cykl PLC), nebo k zbytečnému prodloužení cyklu programu několikanásobným voláním přerušení k výpočtu průběhu (krok generátoru menší než cykl PLC).

Funkce **FC1 – SinGen** obsahuje parametry pro nastavení generátoru a jeho výstup, parametry mají délku slova 16 bitů, se znaménkovým bitem a výpočet se provádí celočíselně, s mezivýpočtem funkce sinus s pohyblivou desetinou čárkou. Parametry pro nastavení jsou amplituda generovaného sinusového signálu, střední hodnota otáček a perioda průběhu. Funkce je doplněna bitovými vstupy pro povolení výstupu, nulování času běhu a spuštění generátoru. Funkce je podrobně popsána v příloze A.1.

2.1.3 Vytváření DB bloků

Vzhledem k možnosti různého množství zaznamenaných hodnot a omezení velikosti integrované paměti v PLC vedlo k vytvoření funkce, která umožňuje vytvoření DB bloku v paměti RAM nebo na kartě MMC, jejíž velikost lze v případě potřeby většího obsahu dat zvětšit. DB bloky v RAM slouží pro dočasné uložení hodnot, které se zapisují v pořadí za sebou. Tímto uspořádáním lze na kartu MMC uložit naměřená data, která by se do RAM nevešly.

Byl vytvořen funkční blok **FB1 – TotCreateDB** k dynamickému vytvoření DB bloku s parametry definující velikost dané DB, v jaké paměti bude umístěn a nastavení čísla DB bloku, pokud daný blok již existuje a souhlasí požadovaná délka, je blok zachován, jinak je smazán a vytvořen nový. Podrobný popis je v kapitole přílohy A.6

2.1.4 Kopírování DB bloků

Při dynamickém vytváření DB bloků nastává problém ve spojení OPC serveru a automatu PLC, kdy je OPC konfigurováno s vazbou na vytvořené DB bloky při psaní programu PLC a nemá tedy informace o dynamicky vytvořených blocích. Tento problém jsem vyřešil vytvořením DB bloku při psaní programu, který má již danou velikost, jež jsem zvolil s ohledem na velikost paměti RAM a kompromisem mezi velikostí a rychlostí kopírování na délku 100 položek typu word. Byla napsaná funkce **FC5 – DB_Copy** využívající systémové funkce (SFC) v PLC, pro kopírování DB. Funkce umožňuje kopírování bloků hodnot mezi DB bloky s nastavením zdroje a cíle kopírování a délky dat. Funkce je navržena tak, že DB bloky mohou být jak v paměti RAM, tak i na kartě MMC, jen je potřeba pomocí parametrů nastavit směr kopírování. Podrobný popis funkce je v kapitole přílohy A.5.

2.1.5 Indexování ukládání a čtení hodnot v tabulce

Pro požadavky časové analýzy na rychlost ukládání dat do tabulky kterou představuje dynamicky vytvořený DB blok, byla vytvořena funkce **FC3 - TimeStmp** pro ukládání hodnot jdoucích za sebou v časové posloupnosti do DB bloků tak, že v každém průběhu programovým cyklem PLC zvýší index řádku na který má být daná hodnota zapsána. Aby nebylo využití funkce omezeno pouze na využití k ukládání hodnot, byla rozšířena o další režimy. V druhém režimu čte hodnotu času, seřazené ve vzestupném pořadí, na daném indexu řádku z daného DB bloku a při překročení s porovnávaným časem běhu přepne na následující index řádku. Tohoto režimu lze využít například při generování průběhu definovaného obsluhou. V třetím režimu je zvyšování indexu řádku řízeno sestupnou hranou řídicího bitu, toho lze využít například pro uložení hodnoty po ustálení při skokové změně žádaného průběhu. Podrobný popis funkce je v kapitole přílohy A.3.

2.1.6 Kopírování dočasných hodnot s MMC kartou

S návrhem řešení ukládání hodnot větších rozsahů na kartu MMC se objevila otázka jak zajistit kopírování hodnot mezi MMC a RAM s minimálním zásahem z uživatelské aplikace ovládající PLC. Po napsání funkce FC3 - TimeStmp pro kopírování DB bloků se rýsovalo řešení, které jsem realizoval pomocí této funkce, doplněné o algoritmy hlídající stav zápisu hodnot do dočasných DB bloků v RAM a aktivující kopírování tak, aby nedošlo k přepsání dočasných hodnot novými před zkopírováním do DB bloku v MMC. Po napsání funkce pro ukládání a čtení hodnot byla funkce pro kopírování s MMC dále rozšířena o kopírování žádaných hodnot v DB bloku na MMC do dočasného DB bloku v RAM, aby bylo možné obě funkce spojeně i v jiných aplikacích. Tuto funkci jsem označil názvem **FC8 – ReqCopy**. Podrobný popis této funkce je v kapitole přílohy A.8.

2.1.7 Ostatní podpůrné bloky

Pro podporu předešlých funkcí bylo vytvořeno dalších pět bloků, jejichž jednotlivý popis by byl příliš krátký a mohl by znepřehlednit osnovu.

Prvními dvěma bloky jsou DB které popisují standardní členění typu PPO 2 pro vysílání dat nazvaný **DB2 – PPO2_Out** a příjem dat nazvaný **DB1 – PPO2_In**. Tyto bloky jsou popsány v kapitolách 8 a 9 přílohy A.

Dalším blokem je funkce **FC7 – TransDB** pro případnou změnu pořadí, s vypuštěním zápisu, ze vstupního DB bloku na výstupní. Popis funkce je v kapitole přílohy A.7.

Posledníma dvěma bloky jsou funkce pro čtení z DB bloku **FC2 – RdFrDB** a zápis do DB bloku **FC4 – WrToDB**, pomocí pevně definované adresy a hodnoty indexu řádky z funkce **FC3 - TimeStmp**. Podrobný popis těchto funkcí je v kapitolách přílohy A.2 a A.4.

2.1.8 Hlavní program

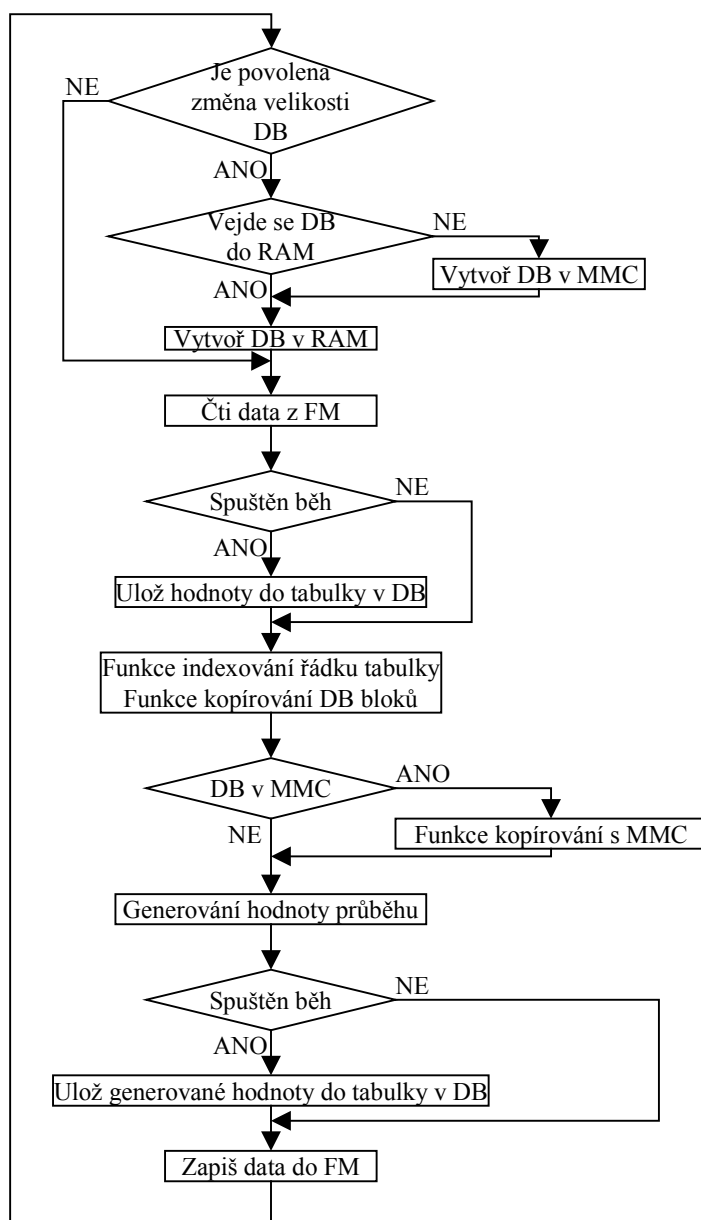
Hlavní program je optimalizován tak, aby při běhu časové analýzy byl cyklus programu proveden v co nejkratším čase a zároveň bylo zamezeno případné změně velikosti dynamických DB bloků, což by mohlo způsobit ztrátu uložených dat. V případě, že požadovaná velikost DB bloků se vejde do paměti RAM, nejsou DB bloky vytvořeny na kartě MMC a v průběhu měření je vynechána funkce pro kopírování s MMC.

Na začátku hlavního programu je testována podmínka jestli je povolena změna velikosti dynamických DB bloků. Při splnění této podmínky je kontrolována délka bloků a podle nastavených proměnných v PLC zjištěno zda se vejde jen do RAM nebo na MMC. Vejdou-li se DB bloky do RAM, je dynamicky vytvořeno potřebné místo pro data. Pokud je požadována větší délka dat než se vejde do RAM, je v RAM vytvořena maximální povolená délka DB bloku a na MMC jsou vytvořeny DB bloky o požadované délce. Při takovém uspořádání slouží DB bloky v RAM k dočasnému uložení hodnot a jsou během měření kopírovány do DB bloků v MMC.

Pro vytvoření DB bloků je volána funkce **FB1 - TotCreateDB**, kterou je nutno volat pro každý vytvářený DB blok. V další části programu jsou čteny hodnoty ze sběrnice PROFIBUS a pak následně, je-li spuštěn běh analýzy, uložení hodnot do DB bloků voláním funkce **FC4 - WrToDB**, tím je zamezeno případnému přepsání hodnot při pozastavení běhu analýzy. Po uložení hodnot je volána funkce **FC3 - TimeStmp** pro indexování čtení a zápisu, po kterém následuje funkce **FC5 – DB_Copy** pro kopírování DB bloků. V případě že jsou DB bloky vytvořené i v MMC, je volána funkce **FC8 - ReqCopy** pro řízení kopírování s MMC. Nyní už je volána funkce **FC1 – SinGen** pro generování žádané hodnoty, která je pak uložena v tabulce voláním funkce **FC4 - WrToDB** a pak jsou potřebná data k řízení FM

poslána k vysílání na jednotlivé frekvenční měniče. Celý algoritmus programu zobrazuje následující obrázek.

Při spuštění PLC se maximální čas cyklu pohybuje přibližně okolo 20 ms, což je způsobeno vytvářením dynamických DB bloků, během běhu programu při spuštění měření časové analýzy se čas jednoho cyklu programu pohybuje v rozmezí 3 – 6 ms, což by měl být při měření dostačující časový krok.

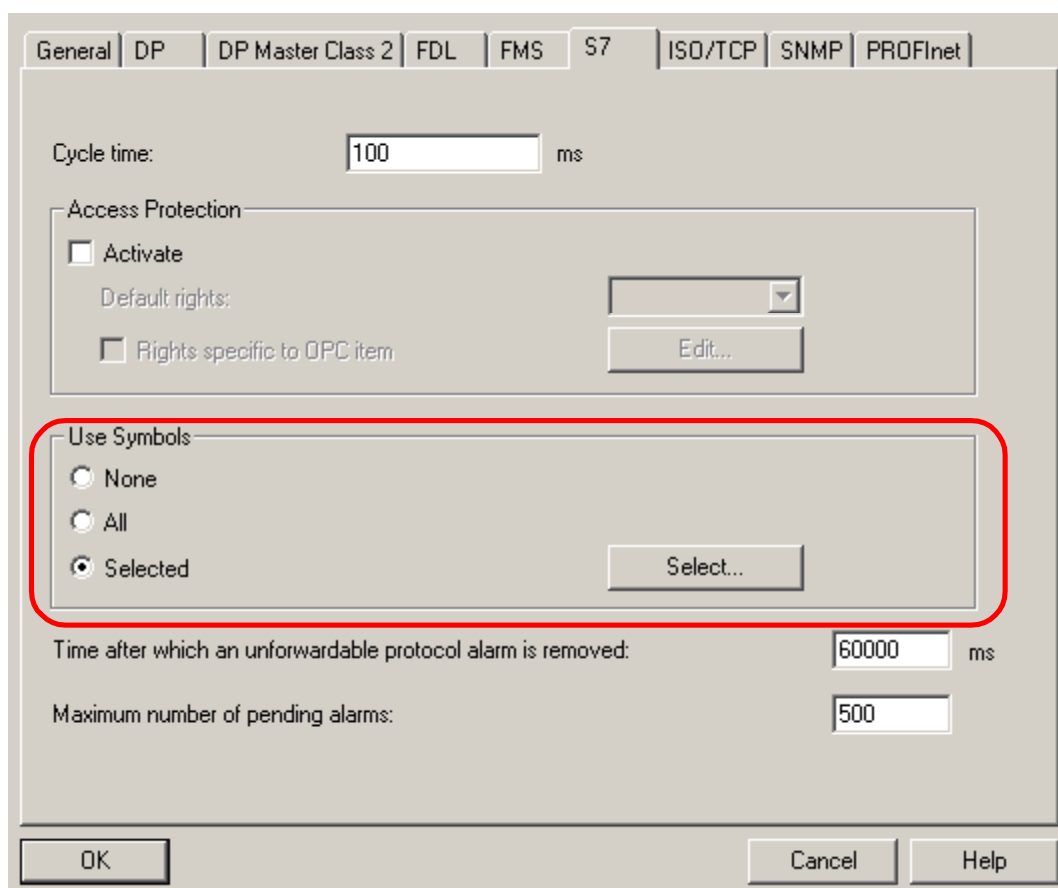


Obr. 2.1.8-1: vývojový diagram programu PLC

2.2. Program a konfigurace OPC

Jak již bylo zmíněno v konfiguraci PLC, je pro správnou komunikaci mezi OPC server a OPC klient potřeba správně nakonfigurovat propojení kanálů mezi zařízeními PLC – OPC server – OPC klient.

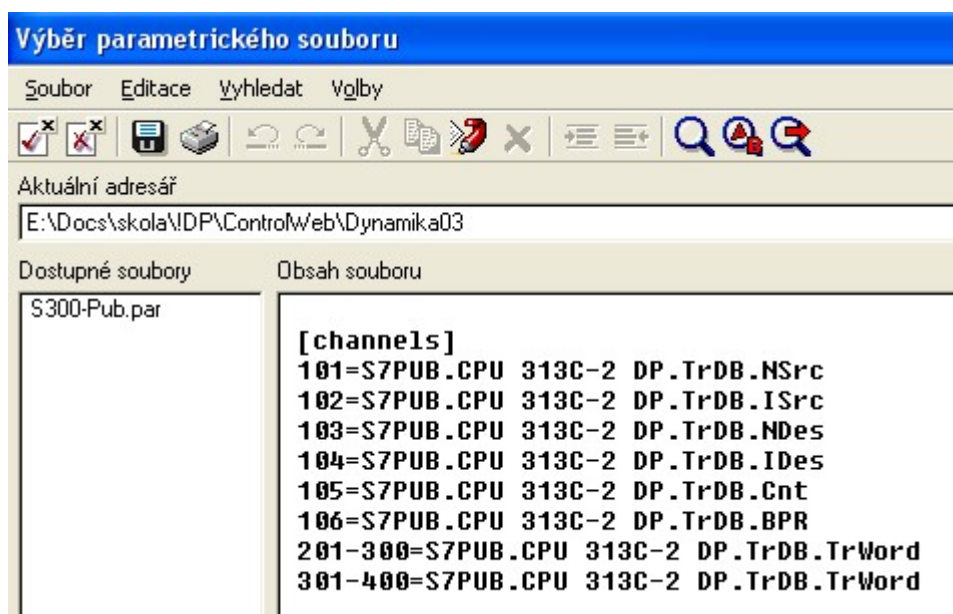
Rozhraní OPC server se nastavuje pomocí vývojového prostředí k nastavení PLC, kde se vytvoří příslušná stanice PC, zobrazená na obrázku Obr.2.1.1-2, s nastavením jaký typ hardware se používá ke komunikaci a verzi spuštěné aplikace zprostředkující OPC server. Nastavení propojení kanálů OPC a PLC není automatické a je potřeba ho nastavit ve vlastnostech OPC, jaké proměnné se mají sledovat, pro ilustraci zobrazení nastavení proměnných ve zvýrazněné oblasti na Obr.2.2-1. Při změně proměnných v programu PLC nedojde k automatické aktualizaci kanálů s PLC a je třeba vazbu na proměnné zrušit, provést kontrolu konfigurace prostředím, vazbu opět nastavit, aby byla aktualizovaná, a nahrát konfiguraci do stanice.



Obr. 2.2-1: Nastavení proměnných pro OPC server

Rozhraní OPC klient je použito ve vývojovém prostředí Control Web, jedná se o objektově orientované prostředí s použitím jazyka Pascal rozšířeného o prostředky

určené pro řízení a vizualizaci automatizace. Způsob práce s rozhraním OPC klient se v různých vývojových prostředí může odlišovat, vzhledem k nezávislosti OPC lze však nastavení vazby s OPC server zobecnit. Pro práci s OPC je potřeba mít příslušný ovladač, který může být dodáván v základu vývojového prostředí nebo je rozšířením prostředí daného výrobce, jak to je v případě Control Web. Pro správnost komunikace OPC musejí být definovány názvy kanálů z OPC serveru a jejich přiřazení ve vývojovém prostředí s OPC klient, nelze však přesně definovat jak je každý kanál pojmenován, to záleží na každém programu OPC server a proto je vhodné pomocí programu, umožňující prohlížení kanálů a jejich názvů v OPC serveru, zjistit příslušné názvy a použít je k vytvoření vazby. Pro ilustraci vytvoření vazby OPC klient-server následuje obrázek zobrazující nastavení názvů kanálů v OPC serveru a jejich přiřazení k číslům kanálů v OPC klient.



Obr.2.2-2: Propojení kanálů OPC klient-server

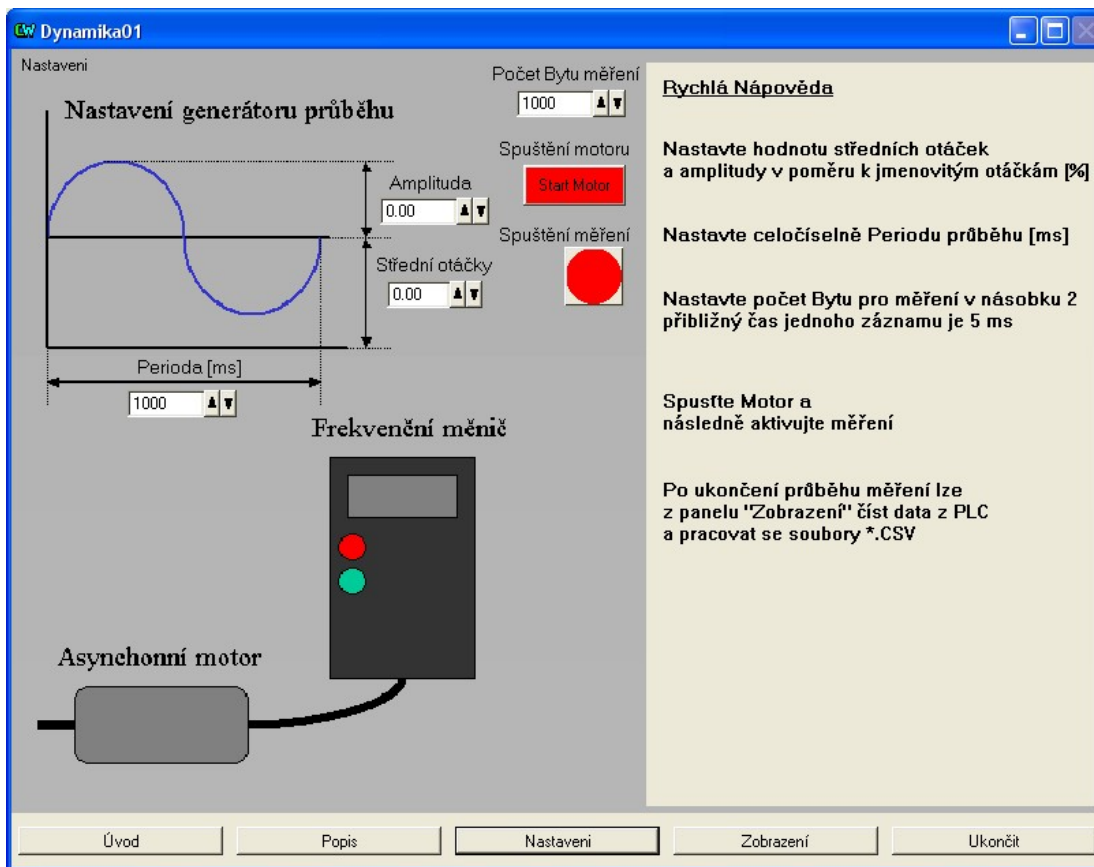
2.2.1 Vývoj programu v Control Web

Vývoj programu byl realizován ve dvou verzích programu, z nichž první verze je čistě pracovní a druhá koncová určena pro potřeby této práce.

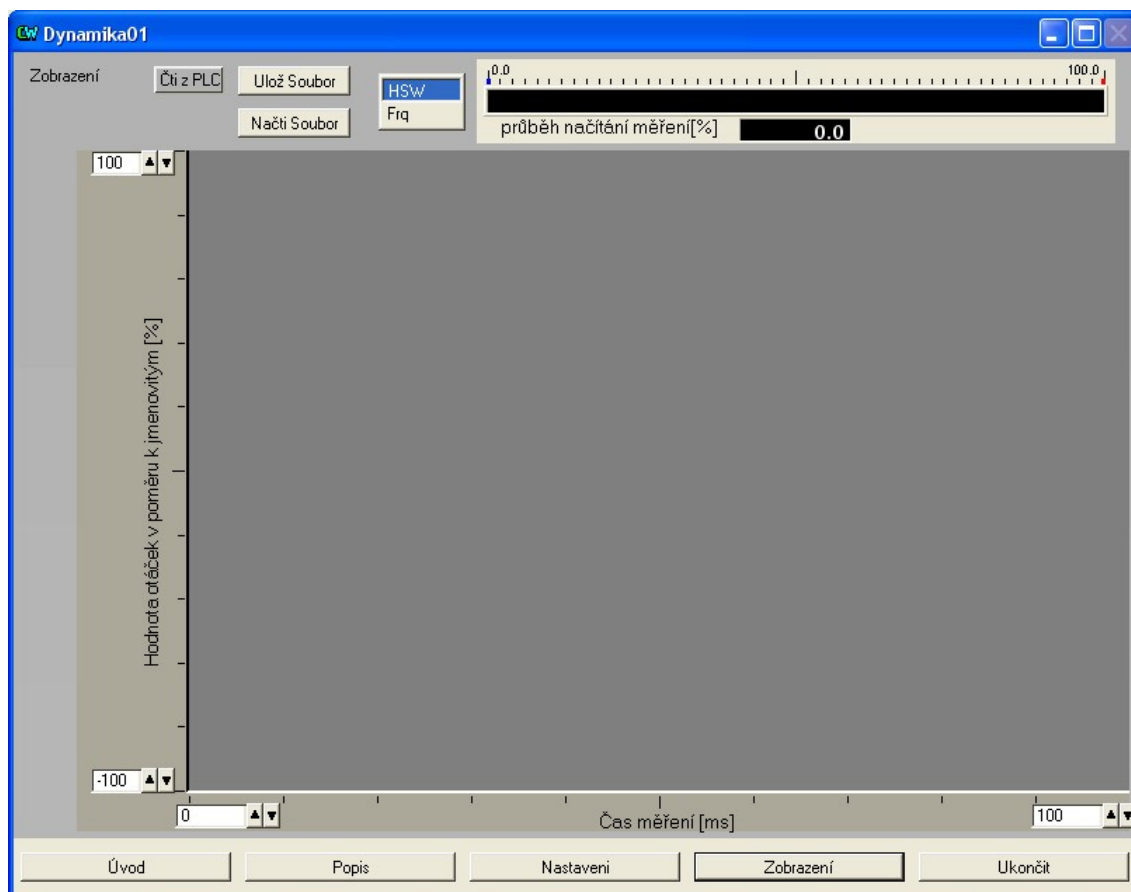
V první verzi bylo vyvinuto aplikování všech funkcí v PLC tak, aby při případném použití vyvinutých funkcí PLC pro obdobné řízení bylo možné použít i vizualizační a řídicí algoritmy z prostředí Control Web a byla tím vytvořena kompletní celek funkcí pro řízení a vizualizaci. O této pracovní verzi se zmiňuji jen okrajově z důvodu předčasného ukončení vývoje, přesto však stojí za zmínění vzhledem k

vytvořeným funkcím. Aplikace je umístěna na přiloženém CD v adresáři „\Aplikace Contol Web\Prac“.

Druhá verze programu využívá potřebných funkcí k řízení a zobrazení měření časové analýzy, aplikaci je možno spustit v režimu řízení v reálném čase, vhodném pro přímé řízení pomalých procesů, nebo v režimu závislosti na změnách dat, vhodného pro vizualizaci a řízení, kde není nutné dodržení reálného času. Vzhledem k protichůdným vlastnostem zařízení a komunikace (malá rychlost reakce OPC a velká rychlost analýzy PLC) jsem zvolil režim běhu aplikace v závislosti na změnách dat. V aplikaci se nacházejí dva oddělené panely, jeden určen k nastavení měření a řízení běhu PLC (Obr. 2.2.1-1), a druhý k vizualizaci a práci s naměřenými daty (Obr. 2.2.1-2). Tímto rozdělením jsem chtěl oddělit část řízení, která nastavuje běh PLC, a část vizualizace, která je závislá při čtení hodnot z PLC na nastavení parametru velikosti záznamu měřených hodnot a s tím souvisejících parametrů v PLC, které je od spuštění běhu analýzy samostatné měřicí zařízení, bez nutnosti zásahu uživatele a samo po dokončení měření zápisem posledních hodnot automaticky ukončí běh. Podrobný popis funkce a ovládání aplikace je v příloze B. Aplikace je na přiloženém CD v adresáři „\Aplikace Contol Web\Dynamika“



Obr. 2.2.1-1: Panel Aplikace k řízení měření průběhů



Obr. 2.2.1-2: Panel aplikace k čtení a ukládání měření

3. Analýza dynamiky řízení

Cílem tohoto měření je určit závislost času obsluhy pro daný frekvenční měnič v závislosti na komunikační rychlosti a počtu řízených měničů a tyto hodnoty pak následně aplikovat ve vytvořeném simulačním modelu. Sběrnice PROFIBUS je sice univerzálním otevřeným rozhraním, ale možnost získání přesného popisu komunikace je zpoplatněna organizací spravující technologii rozhraní, proto analýza bude provedena z měřených dat bez možnosti porovnání s přesným teoretickým rozbohem na základě znalosti podrobné specifikace rozhraní.

Měření bylo prováděno vytvořenou aplikací pomocí OPC, ta po spuštění měření sledovala hodnoty určující průběh měření, což způsobuje zatížení komunikace v některých částech měření, toto zatížení bude třeba zohlednit a případně eliminovat v simulačním modelu.

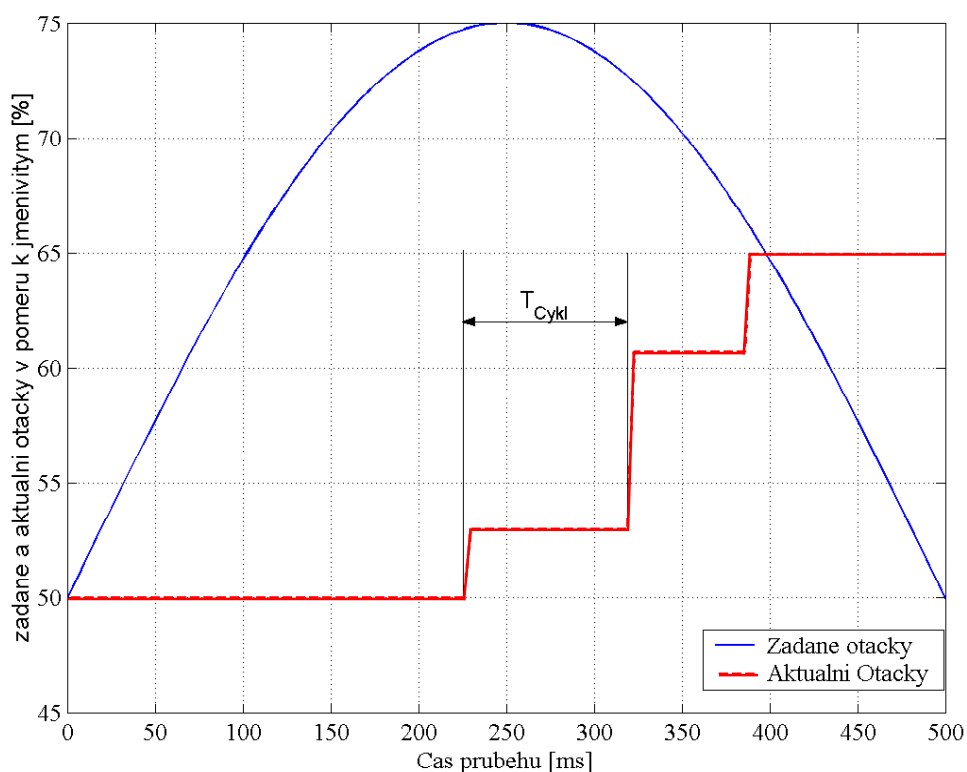
3.1. Analýza komunikace

Komunikace probíhá zároveň s během řídicího programu a je řízena obslužnými funkcemi systému, které před každým cyklem programu zapíší kompletně přijatá data ze sběrnice, čímž umožní čtení nových dat programem, a po skončení cyklu programu jsou nová data připravena k zápisu na sběrnici. Tímto mechanismem dochází při stejném čase jednoho cyklu a různé rychlosti komunikace ke dvěma způsobům časování měření. Při vysoké rychlosti komunikace je cyklus obslužení všech měničů proveden v průběhu jednoho programového cyklu, čímž se přesnost řízení a měření omezuje na rychlost programu, která tak udává minimální časový krok mezi dvěma žádanými hodnotami otáček pro jeden měnič. Při nízké rychlosti komunikace je cyklus obslužení všech měničů proveden v několika cyklech programu, čímž je přesnost řízení dána cyklem komunikace a algoritmus měření zaznamená kdy je dokončen cyklus komunikace.

3.1.1 Teoretický rozbor analýzy komunikace

S přihlédnutím na omezení analýzy komunikace lze z měření s malou přenosovou rychlostí, kdy je zaznamenáno časování cyklu komunikace, zobecnit odvození vztahů pro komunikaci na základě známých parametrů použitých při měření.

Z měření lze zjistit celkový čas cyklu komunikace (T_{Cykl}), který zobrazuje Graf 3.1.1-1 změřený při komunikační rychlosti $C = 19200$ Baud, periodě generovaného signálu $T = 1000$ ms, střední hodnotě otáček $O_{\text{st}} = 50$ %, amplitudě signálu $A = 25$ % a počtem připojených motorů $PM = 1$.

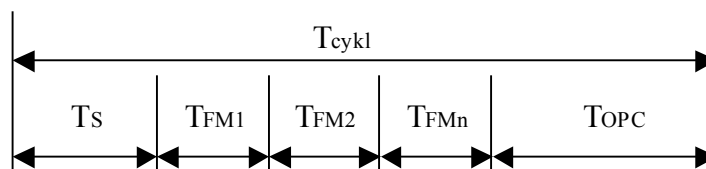


Graf 3.1.1-1: Odečtení cyklu komunikace

$C = 19200$ Baud, $T = 1000$ ms, $O_{st} = 50$ %, $A = 25$ % PM = 1

Celkové zpoždění na začátku průběhu aktuálních otáček je způsobeno dynamikou soustavy FM-motor a možným načasováním komunikace, kdy cyklus komunikace začal s minimálním předstihem před okamžikem spuštění měření.

Čas T_{Cykl} lze rozdělit na menší časové úseky (Obr. 3.1.1-2), které reprezentují čas obsluhy synchronizace sběrnice, čas obsluhy jednotlivých měničů a čas obsluhy komunikace s OPC.



Obr. 3.1.1-2: Rozdělení cyklu komunikace na úseky

Pro rozdělení celkového času cyklu komunikace platí:

$$T_{cycl} = T_S + T_{FM1} + T_{FM2} + \dots + T_{FMn} + T_{OPC} \quad (r. 3.1.1-1)$$

T_{cycl} – celkový čas cyklu komunikace

$T_{FM1} \dots T_{FMn}$ – čas komunikace s jednotlivými měniči

T_{OPC} – čas komunikace s OPC

Za předpokladu stejné délky přenášených parametrů s měniči lze vztah r. 3.1.1-1 upravit na:

$$T_{cycl} = T_S + PM \cdot T_{FM} + T_{OPC} \quad (r. 3.1.1-2)$$

PM – počet připojených FM

T_{FM} – čas komunikace s každým FM

OPC nekomunikuje s PLC v každém komunikačním cyklu a žádost na změnu nebo čtení hodnot může být různě rozsáhlá, proto je čas T_{OPC} velice proměnlivý a není možné ho jednoznačně určit. Z těchto důvodů je pro zjednodušení vztahu r. 3.1.1-2 hodnota T_{OPC} přičtením vnořena do hodnoty T_S a tím zohledněna komunikace, nebo zanedbáním T_{OPC} při vybrání několika nejkratších cyklů T_{cycl} , čímž bude analýza provedena jen pro řízení frekvenčních měničů bez komunikace OPC. Upravený vztah má pak tvar:

$$T_{cycl} = T_S + PM \cdot T_{FM} \quad (r. 3.1.1-3)$$

Z několika měření s různým počtem FM lze pak vypočíst z průměrovaného měření hodnoty T_{cycl} vztahem r. 3.1.1-3 čas potřebný ke komunikaci s jedním FM. Při znalosti komunikační rychlosti a délky řídících dat lze určit objem dat určených k řízení komunikace, který při změně délky řídících dat FM zůstává stejný. Pro výpočet délky dat k řízení komunikace platí vztah:

$$N_K = \frac{\left[\frac{T_{FM} \cdot C}{N_B} - 2 \cdot N_R \right]}{2} \quad (r. 3.1.1-4)$$

N_K – počet Bytů řídících komunikaci

C – komunikační rychlost sběrnice

N_B – počet bitů na přenos jednoho Bytu

N_R – počet Bytů pro řízení FM (procesní data, délka dle PPO typu)

3.1.2 Změření a analýza dynamiky komunikace

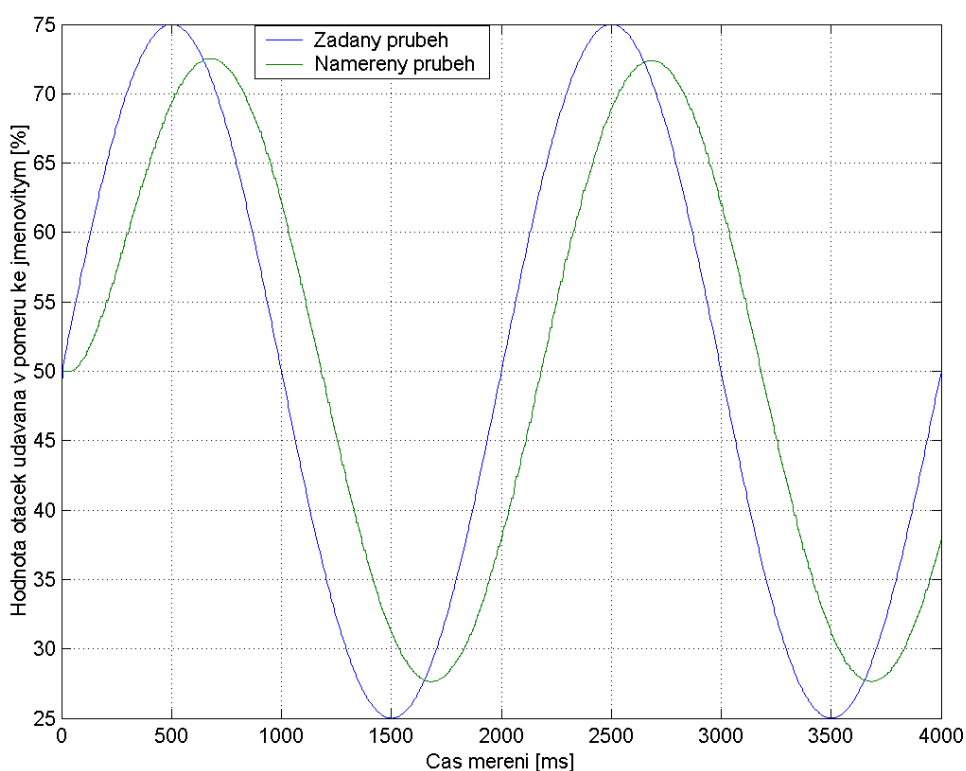
Pro potřeby analýzy a ověření vytvořeného simulačního modelu řízení byli naměřeny průběhy otáček motoru s těmito parametry:

Počet frekvenčních měničů PM = 1; 2; 4

Komunikační rychlost C = 19200; 45450; 187500; 1500000 Baud

Perioda generovaného průběhu $T = 500; 1000; 2000; 3000; 4000; 5000$ ms

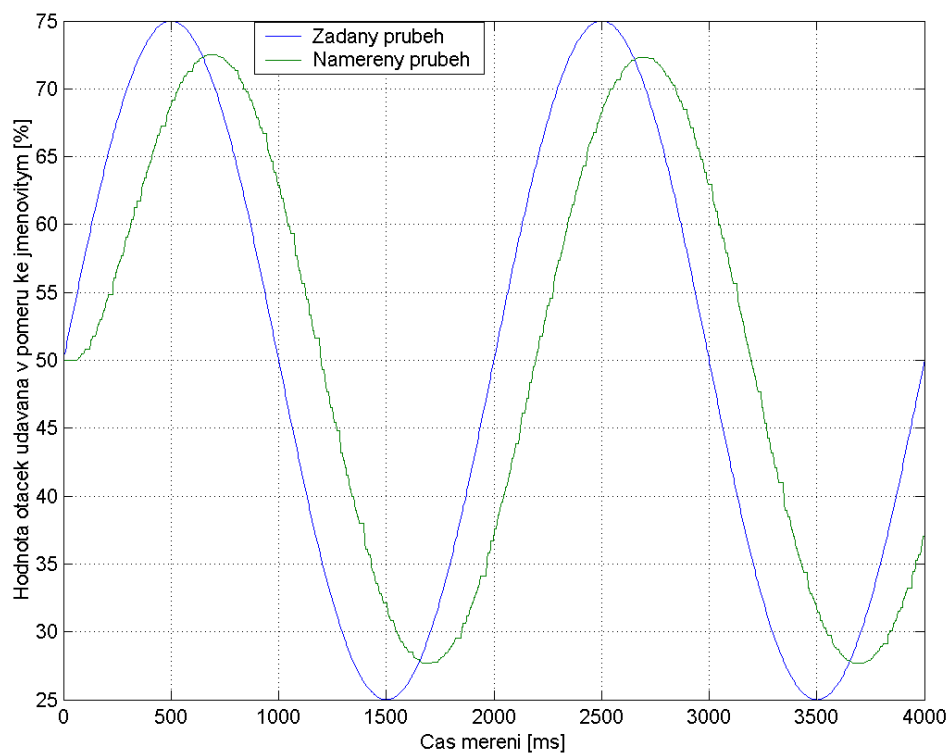
Z velkého množství kombinací parametrů měření vznikl velký soubor dat pro analýzu, která by při jednotlivých měření byla zdlouhavá a mohla by dát velmi rozdílné výsledky, které by pak stejně bylo potřeba porovnat a vybrat vhodnou skupinu jako referenci pro výpočet vztahů z kapitoly 3.1.1, definující vlastnosti komunikace. Při stejné přenosové rychlosti komunikace a stejném počtu FM je celkový čas cyklů komunikace, při zanedbání času komunikace s OPC, přibližně shodný, proto jsem se rozhodl sloučit data měření se souhlasným počtem FM a rychlostí komunikace, a tím rozšířit počet hodnot pro statistické výpočty. Pro ukázkou zobrazuji grafy průběhů měření pro každou přenosovou rychlost se shodujícími se ostatními parametry. Výpis jednotlivých parametrů je uveden pod příslušným grafem.



Graf 3.1.2-1: zobrazení měření pro $C = 1,5$ MBaud

Komunikační parametry: $C = 1,5$ MBaud, $PM = 1$

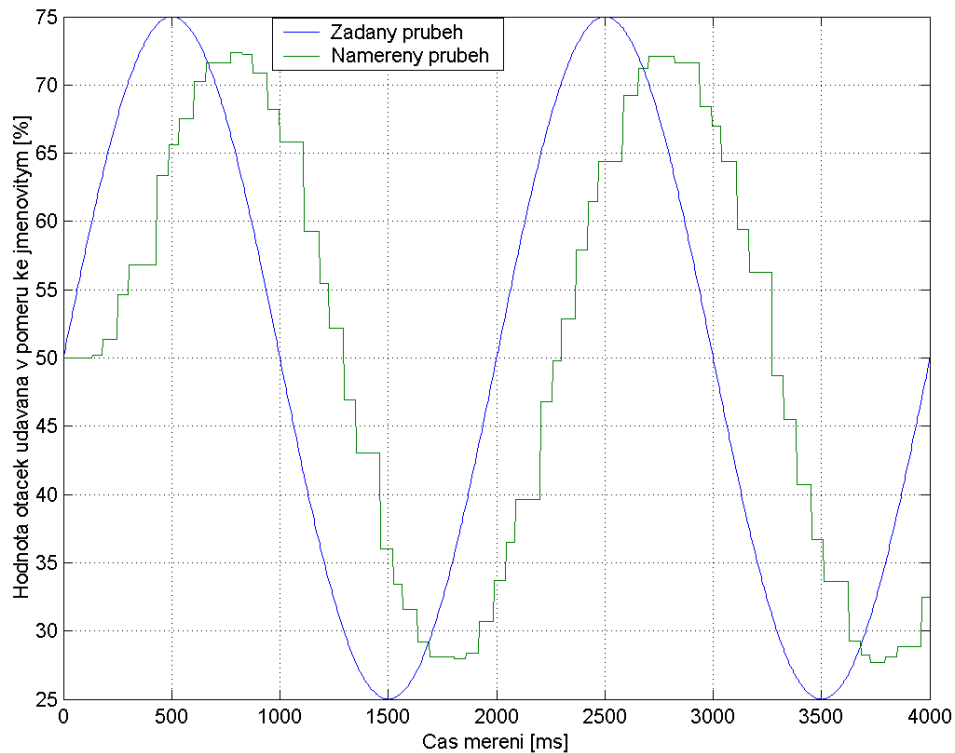
Parametry Generátoru: $Ostr=50\%$, $A=25\%$, $T=2000$ ms



Graf 3.1.2-2: zobrazení měření pro $C = 187,5$ Kbaud

Komunikační parametry: $C = 187,5$ kbaud, $PM = 1$

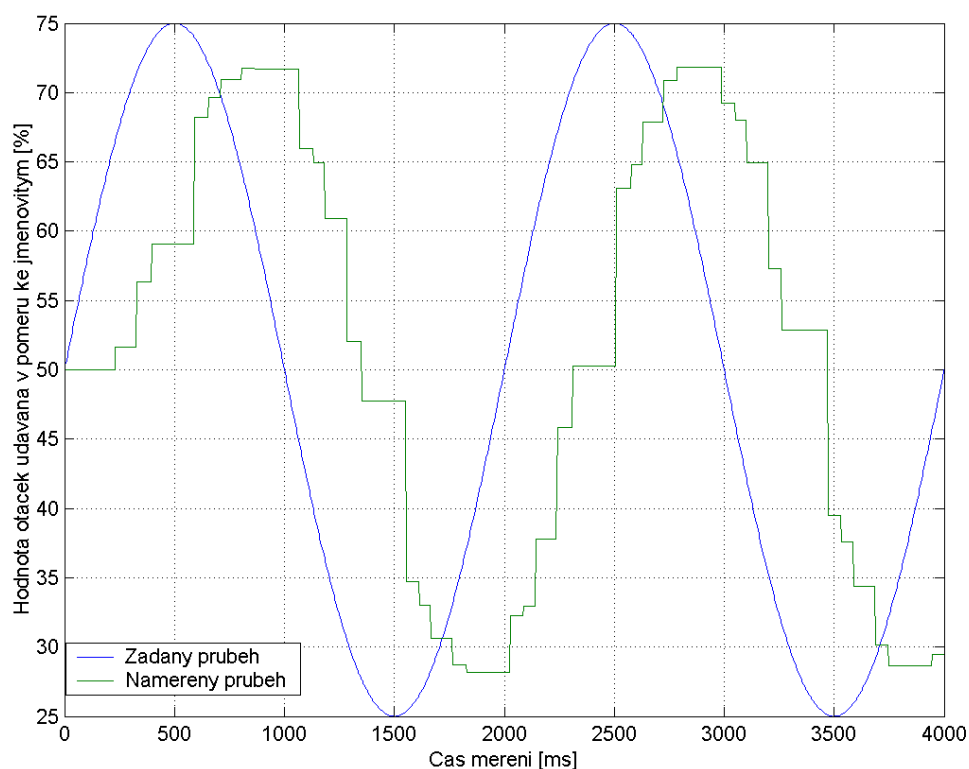
Parametry Generátoru: Ostr=50%, A=25%, T=2000 ms



Graf 3.1.2-3: zobrazení měření pro $C = 45,45$ Kbaud

Komunikační parametry: $C = 45,45$ kbaud, $PM = 1$

Parametry Generátoru: Ostr=50%, A=25%, T=2000 ms



Graf 3.1.2-4: zobrazení měření pro $C = 19,2 \text{ KBaud}$

Komunikační parametry: $C = 19,2 \text{ kBaud}$, $PM = 1$

Parametry Generátoru: $Ostr=50\%$, $A=25\%$, $T=2000 \text{ ms}$

Pro nalezení referenční hodnoty času cyklu komunikace pro další výpočty jsem vytvořil skript v programu MATLAB 6, který procházel jednotlivé naměřené průběhy otáček a při skokové změně hodnoty zaznamenal rozdíl času od předchozího skoku. Pak jsou hodnoty časů skoku u měření se stejnou hodnotou komunikační rychlosti a počtem FM spojeny. Pro výběr hodnot bylo potřeba nalézt nejvhodnější statistickou funkci s případnou selekcí hodnot hodně se lišících, protože komunikace na sběrnici je zatěžována mimo řízení FM také aktualizací proměnných v OPC serveru a tato komunikace nelze časově řídit ani zaznamenat její přístup na sběrnici. Po několika vyzkoušených funkcí a porovnání linearitu nárůstu času pro více FM jsem zvolil výběr hodnoty metodami výpočtu střední hodnoty. Vypočtené hodnoty jsou v následující tabulce 3.1.2-1

Tab. 3.1.2-1: Zjištěné hodnoty cyklu komunikace T_{Cykl}

		Počet připojených frekvenčních měničů		
		Střední hodnota Cyklu komunikace T_{cykl} [ms]		
		1	2	4
Přenosová rychlost [kBaud]	19,2	69,71	104,96	179,87
	45,45	55,46	70,68	107,45
	187,5	9,21	10,18	17,46
	1500	8,22	8,27	10,42

Pro výpočet hodnoty T_{FM} lze ze vztahu r. 3.1.1-3 odvodit

$$T_{cykl1} - T_{cykl2} = T_S + PM_1 \cdot T_{FM} - (T_S + PM_2 \cdot T_{FM})$$

$$T_{FM} = \frac{T_{cykl1} - T_{cykl2}}{PM_1 - PM_2}$$

pomocí toho vztahu lze pak z tab. 3.1.2-1 vypočítat odečtením sousedních sloupců hodnotu T_{FM} , hodnoty jsou v následující tabulce 3.1.2-2

Tab. 3.1.2-2: Zjištěné hodnoty času komunikace s jedním FM T_{FM} [ms]

		Vypočtený čas T_{FM} [ms] přepočtený na $PM = 1$	
		Střední hodnota	
		PM=1	PM=2
Přenosová rychlost [kBaud]	19,2	35,25	36,48
	45,45	15,00	15,20
	187,5	0,97	3,64
	1500	0,05	1,07

Z tabulky 3.1.2-2 je vidět velký rozdíl časů T_{FM} v řádku pro $C = 187,5$ a 1500 kBaud, což je nejspíše způsobeno blízkou hodnotou s časem T_P a tím velkým ovlivněním časem cyklu programu, proto pro výpočet průměrné hodnoty T_{FM} použijí časy pro $C = 19,2$ a 45,45 kBaud. Po přepočtu na normalizovanou $C = 19,2$ kBaud, abychom se zbavili závislosti na různé rychlosti přenosu, jsou hodnoty zprůměrovány, výsledný čas pro komunikaci s FM, pro $C = 19,2$ kBaud, vychází $T_{FM} = 35,8$ ms.

Ze znalosti času komunikace s FM (T_{FM}) lze ze vztahu r. 3.1.1-4 určit počet Bytů řídicích komunikaci

$$N_K = \frac{\left[\frac{T_{FM} \cdot C}{N_B} - 2 \cdot N_R \right]}{2} \quad (\text{r. 3.1.1-4})$$

$C = 19200$ Baud $N_B = 11,5$ $N_R = 20$ (PPO 2 s délkou procesních dat 20 Bytů)

Ze vztahu vychází na obsluhu řízení FM $N_K = 10$ Bytů, tato hodnota je proveditelná a bude použita v další části práce. Pro zhodnocení dosažených výsledků časování komunikace s měniči jsou v následující tabulce zobrazeny teoreticky vypočítané časy pro daný počet měničů při dané komunikační rychlosti se zanedbáním času komunikace s OPC.

Tab. 3.1.2-3: časy obsluhy T_{cykl} [ms] daného počtu FM při dané C

	Čas cyklu komunikace T_{Cykl} [ms]					
	Komunikační rychlost C [kBaud]					
PM	19,2	45,45	187,5	500	1500	12000
1	69,479	29,351	7,115	2,668	0,889	0,111
2	105,417	44,532	10,795	4,048	1,349	0,169
3	141,354	59,714	14,475	5,428	1,809	0,226
4	177,292	74,895	18,155	6,808	2,269	0,284
5	213,229	90,077	21,835	8,188	2,729	0,341
6	249,167	105,259	25,515	9,568	3,189	0,399
7	285,104	120,440	29,195	10,948	3,649	0,456
8	321,042	135,622	32,875	12,328	4,109	0,514
9	356,979	150,803	36,555	13,708	4,569	0,571
10	392,917	165,985	40,235	15,088	5,029	0,629
11	428,854	181,166	43,915	16,468	5,489	0,686
12	464,792	196,348	47,595	17,848	5,949	0,744
13	500,729	211,529	51,275	19,228	6,409	0,801
14	536,667	226,711	54,955	20,608	6,869	0,859
15	572,604	241,892	58,635	21,988	7,329	0,916
16	608,542	257,074	62,315	23,368	7,789	0,974
17	644,479	272,255	65,995	24,748	8,249	1,031
18	680,417	287,437	69,675	26,128	8,709	1,089
19	716,354	302,618	73,355	27,508	9,169	1,146
20	752,292	317,800	77,035	28,888	9,629	1,204
21	788,229	332,981	80,715	30,268	10,089	1,261
22	824,167	348,163	84,395	31,648	10,549	1,319
23	860,104	363,344	88,075	33,028	11,009	1,376
24	896,042	378,526	91,755	34,408	11,469	1,434
25	931,979	393,707	95,435	35,788	11,929	1,491
26	967,917	408,889	99,115	37,168	12,389	1,549
27	1003,854	424,070	102,795	38,548	12,849	1,606
28	1039,792	439,252	106,475	39,928	13,309	1,664
29	1075,729	454,433	110,155	41,308	13,769	1,721
30	1111,667	469,615	113,835	42,688	14,229	1,779
31	1147,604	484,796	117,515	44,068	14,689	1,836
32	1183,542	499,978	121,195	45,448	15,149	1,894

Z tabulky je viditelná velká rozmanitost času cyklu komunikace s měniči, je však otázkou v dané aplikaci řízení zda bude tato rychlost plně využita. Je to způsobeno mechanismem posílání nových hodnot na sběrnici, které je prováděno na konci cyklu programu. Proto je potřeba mimo zhodnocení komunikace navíc přihlídnout k délce cyklu programu. Zjištění časů programového cyklu jsem provedl z časových údajů zaznamenaných při měření v každém cyklu programu získáním změny času mezi po sobě jdoucími hodnotami, tyto změny pak byly průměrovány po skupinách rozdělených podle počtu měničů a komunikační rychlosti. Jednotlivé průměrné časy cyklu programu při nastavených parametrech řízení zobrazuje následující tabulka.

Tab. 3.1.2-4: časy cyklu programu T_P [ms] pro dané nastavení komunikace

Komunikační rychlost [Baud]	Počet připojených měničů		
	1	2	4
19200	4,00	6,11	10,15
45450	4,14	6,14	10,19
187500	4,28	6,34	10,34
1500000	4,28	6,34	10,38

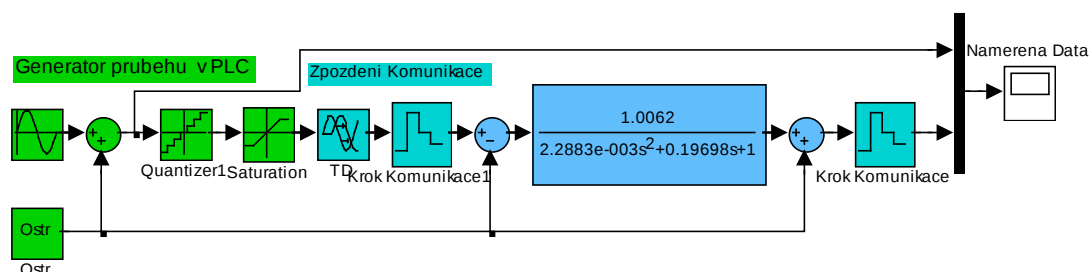
Z uvedené tabulky je vidět nezávislost doby cyklu programu na komunikační rychlosti, malé rozdíly časů jsou způsobeny množstvím analyzovaných dat, naproti tomu je výrazná závislost délky cyklu na počtu řízených FM, to je způsobeno především programem, v kterém je řízení prováděno jedním algoritmem pro různý počet měničů pro které jsou vytvořeny samostatné DB bloky ke komunikaci. Při porovnání s tabulkou 3.1.2-3 je patrné, že pro komunikační rychlost 1,5 MBaud je čas komunikace pro měřený počet měničů kratší než čas cyklu programu, což bude potřeba zohlednit v simulačním modelu.

3.1.3 Vytvoření simulačního modelu

Pro potřeby další práce na analýze dynamiky a zobecnění výsledků bude vytvořen model způsobu řízení, který umožní ověřit dříve zmíněné teorie a výpočty, zajistí po doplnění dalšími funkcemi podporu výpočtů, a simulací ukáže případné omezení kvality řízení pro obecné použití bez nutnosti fyzicky sestavovat dané zařízení. Vzhledem k osobním zkušenostem a praxí bude program vytvořen v simulačním prostředí MATLAB 6.0, které je dostatečně vybaveno simulačními a výpočetními prostředky pro tuto práci.

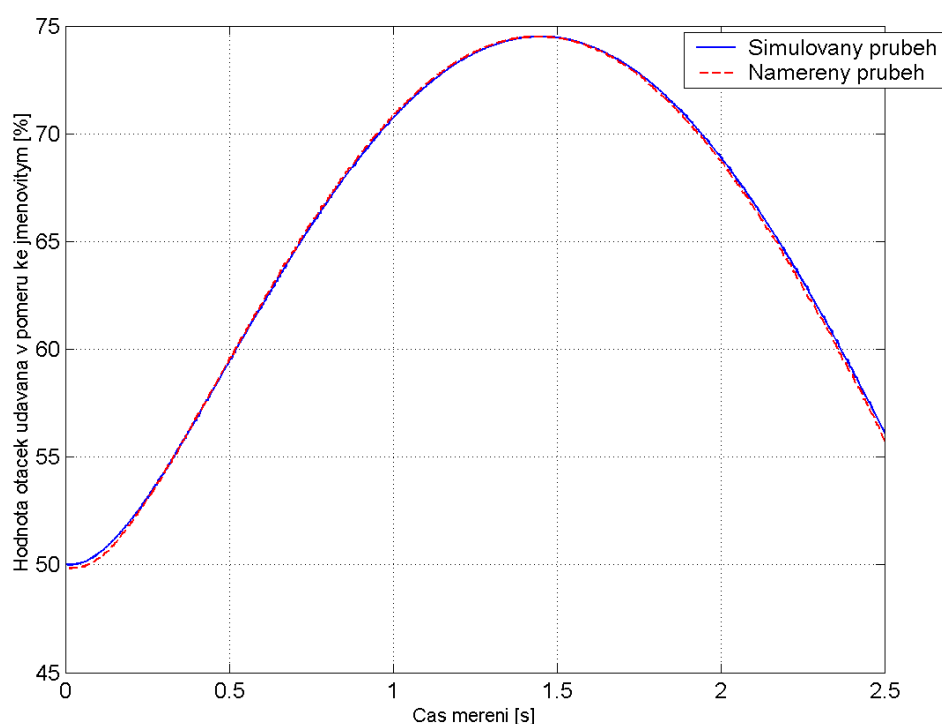
Ze vztahu r. 3.1.1-4 a příslušných parametrů bude vytvořen výpočet pro model komunikace, který doplním o nastavení doby cyklu programu a průměrnou hodnotu

počtu Bytů použitých při komunikaci s OPC. Z naměřených průběhů analýzy komunikace bude identifikací vytvořen model přenosu FM představující akční a měřicí model celku FM – motor. Program pro výpočet modelu a vytvořený model, zobrazen na obr. 3.1.3-1, je uložen na příloženém CD v adresáři „Matlab“. Pro přenos soustavy byla vypočítána mezní frekvence $f_0=0,866$ Hz pro pokles o 3 dB.



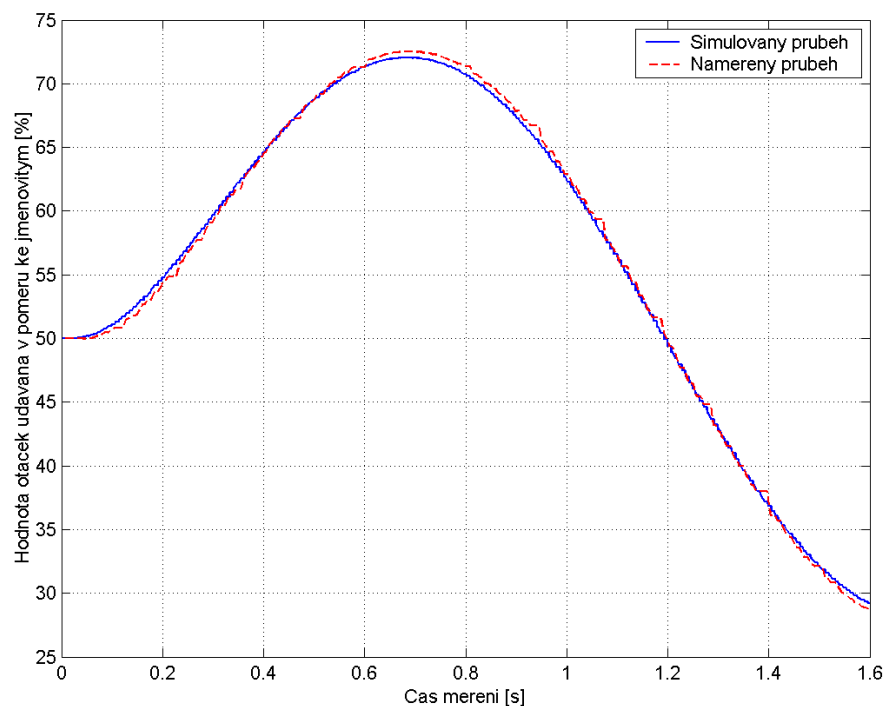
Obr. 3.1.3-1: Model způsobu řízení

Identifikace byla provedena na měření s parametry $C = 1,5$ Mbaud, $PM = 1$ a $T = 5000$ ms a následně byla ověřena správnost modelu porovnáním s nastavením jiného měření. Kontrolu modelu ukazují následující grafy s popsanými parametry nastavení průběhu.



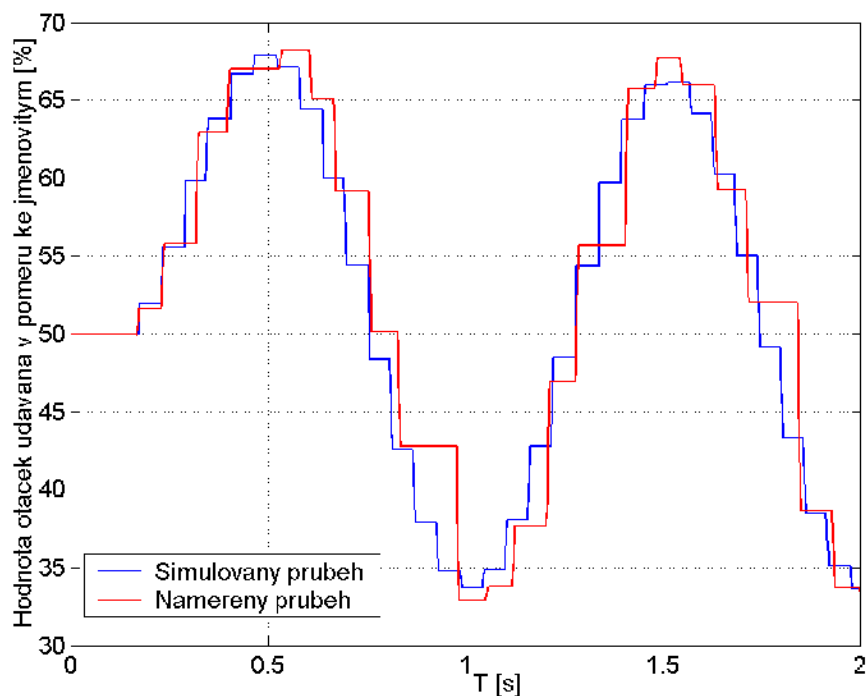
Graf 3.1.3-2: výřez průběhu k identifikaci

$C = 1,5$ Mbaud, $PM = 1$, $T = 5$ s, $O_{str} = 50$ %, $A = 25$ %



Graf 3.1.3-3: Výřez průběhu ke kontrole identifikace

$C = 187,5 \text{ kBaud}$, $PM = 1$, $T = 2 \text{ s}$, $O_{\text{str}} = 50 \%$, $A = 25\%$



Graf 3.1.3-4: Výřez průběhu ke kontrole identifikace

$C = 45,45 \text{ kBaud}$, $PM = 2$, $T = 1 \text{ s}$, $O_{\text{str}} = 50 \%$, $A = 25\%$

Kvalita simul. průběhu=82,05 % Kvalita skutečného průběhu = 82,11 %

Ze zobrazených grafů je zřejmé, že identifikace byla provedena v pořádku a model tak lze použít v dalších částech práce – přestože na grafu 3.1.3-4 nejsou průběhy tak shodné, to je způsobeno zatížením měření komunikací OPC s PLC a lze to zanedbat, vzhledem k tomu že simulovaný průběh více odpovídá sinusovému signálu.

3.2. Posouzení kvality řízení

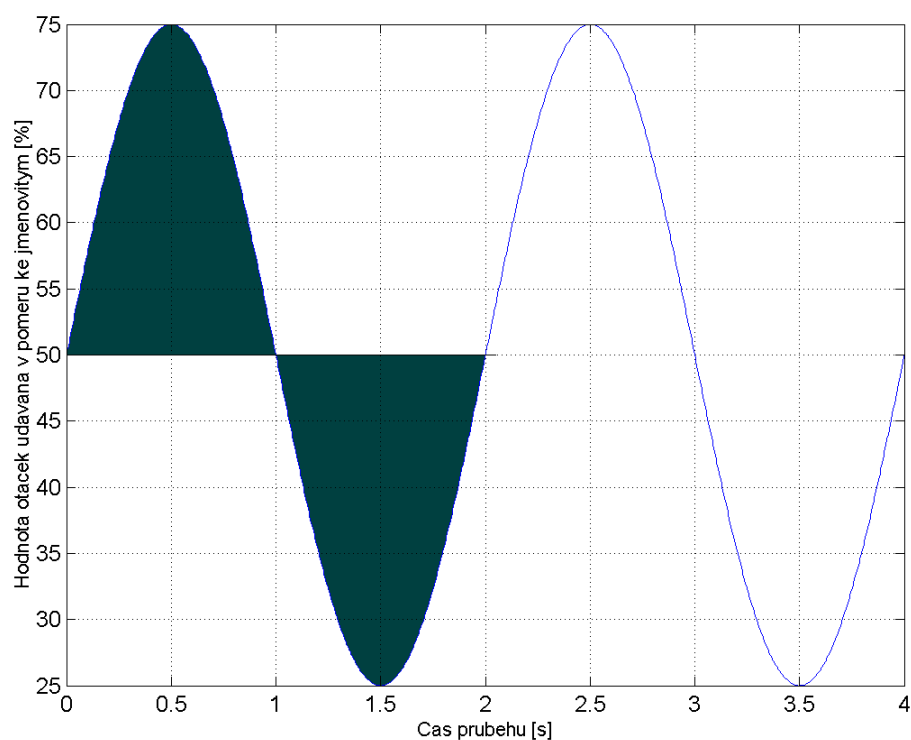
Pro posouzení kvality řízení je třeba definovat jakostní kritérium porovnávající generovaný a skutečný průběh otáček. Kritérium je definováno pomocí ploch mezi křivkami generovaného a skutečného průběhu zobrazených na obr.3.2-1 a obr.3.2-2.

$$Kvalita = \left(1 - \frac{S_R}{S_G} \right) \cdot 100[\%] \quad (\text{r. 3.2-1})$$

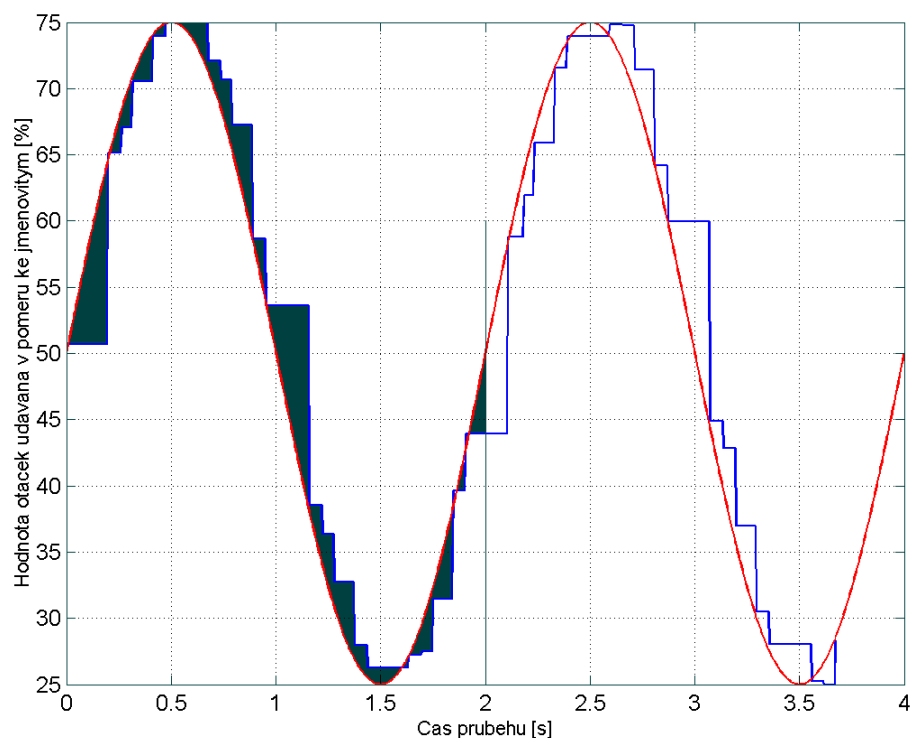
S_R – plocha vzniklá mezi křivkami generovaného a skutečného průběhu

S_G – plocha vzniklá ohraničením průběhem generovaného průběhu a hodnotou středních otáček.

Obecně lze kritérium použít na libovolně dlouhý úsek průběhu, pro tyto výpočty jsem zvolil časový úsek jedné periody generovaného signálu.



Obr. 3.2-1: Určení plochy S_G



Obr. 3.2-2: Určení plochy S_R

Pro plochy průběhů pak platí:

$$S_G = \int |A \cdot \sin(2\pi f t)| dt = \sum_{i=1}^N \left[A \cdot \sin\left(\frac{2\pi f T}{N} i\right) \frac{T}{N} \right] = \frac{AT}{N} \sum_{i=1}^N \sin\left(\frac{2\pi f T}{N} i\right) \quad (\text{r. 3.2-2})$$

$$\begin{aligned} S_G &= \int |A \cdot \sin(2\pi f t) + O_{str} - n_{sim}(t)| dt = \sum_{i=1}^N \left[A \cdot \sin\left(\frac{2\pi f T}{N} i\right) \frac{T}{N} + O_{str} - n_{sim}\left(\frac{iT}{N}\right) \frac{T}{N} \right] = \\ &= \frac{T}{N} \sum_{i=1}^N \left[A \cdot \sin\left(\frac{2\pi f T}{N} i\right) + O_{str} - n_{sim}\left(\frac{iT}{N}\right) \right] \quad (\text{r. 3.2-3}) \end{aligned}$$

A – amplituda generovaného průběhu

f (T) – frekvence (perioda) generovaného průběhu

N – počet vzorků v periodě

Pomocí vztahů r. 3.2-2 a r. 3.2-3 lze pak vyjádřit vztah r. 3.2-1 vyjadřující matematický popis kvality generovaného průběhu otáček, který bych rád v dalších kapitolách nahradil vzorcem obecným, bez nutnosti použití simulačního modelu.

$$Kvalita = \left(1 - \frac{S_R}{S_G} \right) \cdot 100 = \frac{\sum_{i=1}^N \left[A \cdot \sin\left(\frac{2\pi f T}{N} i\right) + O_{str} - n_{sim}\left(\frac{iT}{N}\right) \right]}{\sum_{i=1}^N \left[A \cdot \sin\left(\frac{2\pi f T}{N} i\right) \right]} \quad (\text{r.3.2-4})$$

3.2.1 Výpočet kritéria Kvality

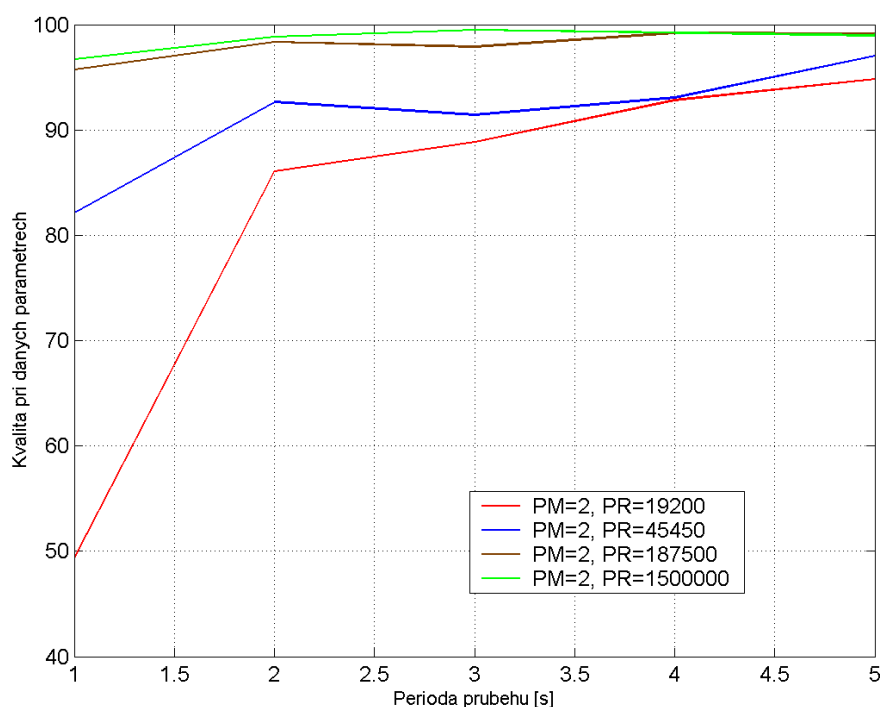
Ze vztahu r. 3.2-4 vyplývá nezávislost samotného kritéria na periodě průběhu T , závislost kvality je dána periodou průběhu T a časem cyklu komunikace T_{Cykl} . Pro posouzení kvality průběhu jsem vytvořil program, který po normalizaci průběhu odstraní případný časový posun a útlum amplitudy, což je způsobeno dynamikou řízení a měření, provede výpočet ploch a kvality průběhu. Program je uložen na přiloženém CD v Adresáři Matlab. Vybrané vypočtené hodnoty kvality jsou v následující tabulce.

Tab. 3.2-1: Vypočtené Hodnoty kvality průběhů

PM	C [kBaud]	T [s]	Kvalita skutečného průběhu [%]
1,00	19,20	2,00	79,05
1,00	19,20	3,00	86,07
1,00	45,45	2,00	92,45
1,00	45,45	4,00	96,04
1,00	187,50	3,00	99,40
1,00	187,50	4,00	98,98
1,00	1500,00	4,00	99,32
1,00	1500,00	5,00	99,58
2,00	19,20	3,00	88,89
2,00	19,20	4,00	92,82
2,00	45,45	2,00	92,71
2,00	45,45	3,00	91,45
2,00	187,50	4,00	99,22
2,00	187,50	5,00	99,18
2,00	1500,00	2,00	98,84
2,00	1500,00	3,00	99,48
4,00	19,20	3,00	86,32
4,00	19,20	4,00	87,64
4,00	45,45	3,00	92,99
4,00	45,45	4,00	93,64
4,00	187,50	2,00	96,40
4,00	187,50	4,00	99,22
4,00	1500,00	3,00	98,74
4,00	1500,00	4,00	98,51

Z uvedených hodnot (Tab. 3.2-1) a následujícího grafu (Graf 3.2-3) je vidět závislost kvality, která však neodpovídá zcela přesně předpokladu se zmenšujícími se nároky na komunikaci se bude zvyšovat kvalita a konvergovat ke 100%. To je především způsobeno zatížením komunikace na sběrnici různým časovým přístupem OPC

serveru, proto nebudu měření provádět na skutečném zařízení, pouze provedu potřebné simulace na modelu a pak porovnáím s naměřenými hodnotami.



Graf 3.2-3: Závislost Kvality naměřených průběhů ne periodě T

3.2.2 Obecné určení kvality

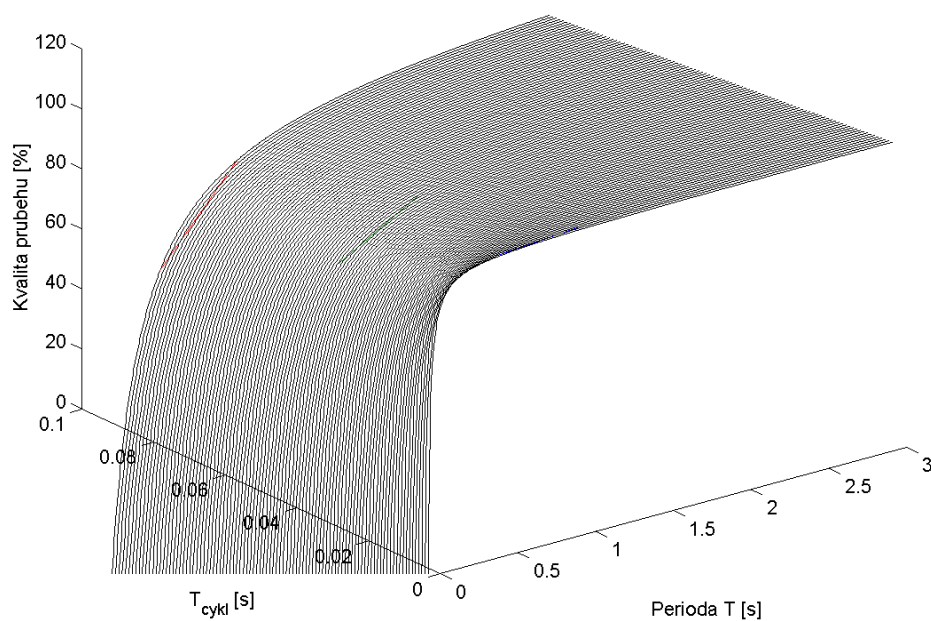
V této kapitole se budu zabývat zjištěním obecné funkce nejlépe popisující kvalitu skutečného průběhu. Vzhledem k předpokládaným vlastnostem průběhu a vztahem mezi periodou průběhu T a časem cyklu komunikace T_{Cykl} jsem zvolil funkci pro obecný výpočet kvality ve tvaru

$$Kvalita = K_1 + K_2 \cdot T_{\text{Cykl}} + \frac{K_3 + K_4 \cdot T_{\text{Cykl}}}{T} [\%] \quad (\text{r. 3.2.2-1})$$

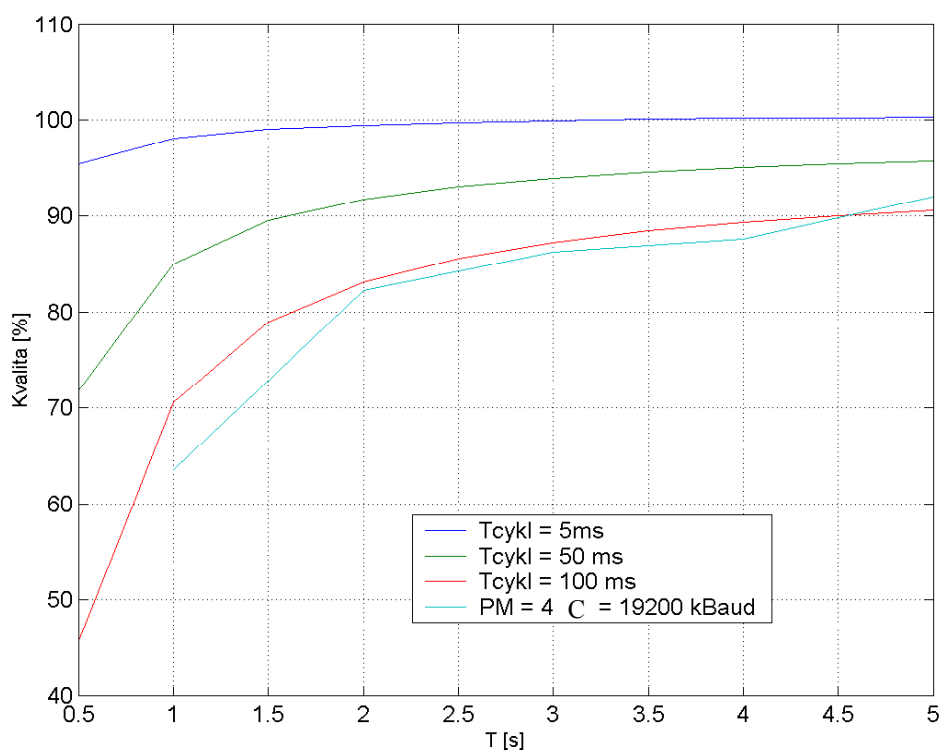
Podle mého názoru je tato funkce nejlépe reprezentující popis kvality, protože se shoduje s vlastnostmi průběhu (Graf 3.2-3). Vztah by měl být obecně použitelný pro jakýkoliv druh generovaného signálu vzhledem k vlastnostem kvalitativního hodnocení.

Pro výpočet byl doplněn program pro nastavení modelu tak, aby bylo simulováno několik průběhů pro různá nastavení periody průběhu T a cyklu komunikace T_{Cykl} , pro tyto nasimulované průběhy byly vypočteny hodnoty kvality, čímž vznikl třírozměrný graf s osami x - perioda T , y - časem cyklu komunikace a z - vypočítanou kvalitou simulovaného signálu (Graf 3.2-4) pro přehlednost jsou na

grafu 3.2-5 jsou zobrazeny průběhy kvality pro 5, 50 a 100 ms s porovnáním se skutečným průběhem.



Graf 3.2-4: Plocha závislosti kvality na T a T_{Cykl}



Graf 3.2-5: Porovnání Kvality skutečného a simulovaného průběhu

Pomocí metody nejmenších čtverců byli vypočítané parametry podle kterých má pak vztah (r. 3.2.2-1) tvar

$$Kvalita = 101,13 - 55,49 \cdot T_{Cykl} + \frac{-1,5672 - 234,4 \cdot T_{Cykl}}{T} [\%] \quad (r. 3.2.2-2)$$

Vztah lze obecně použít pro výpočet kvality průběhu na základě znalosti času periody průběhu a předpokládaného času cyklu řízení, který je možno vypočítat ze vztahů pro určení času cyklu T_{Cykl} uvedených v kapitole 3.1.1.

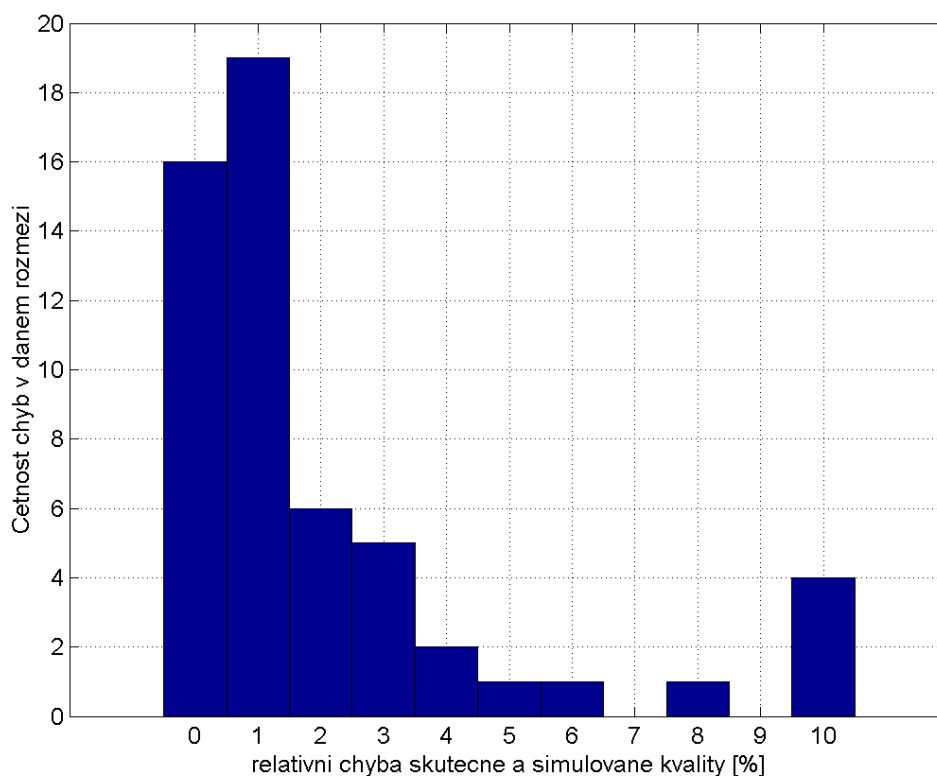
3.3. Ověření modelu a vztahu kvality

Simulační Model již byl ověřen z pohledu přesnosti shodnosti simulovaného řízeného průběhu porovnáním v grafu (Graf 3.1.3-2), nyní když už je metoda k určení kvality bude ještě ověřeno zda simulační model odpovídá i ve shodě se skutečnou hodnotou kvality. Vzhledem k tomu že kvalita skutečného průběhu kolísá a je snížena vlivem komunikace PLC s OPC při porovnávání naměřené hodnoty s modelem, který simuluje pouze komunikaci s FM nebo pro simulaci komunikace s OPC používá střední hodnotu přenášených dat, může docházet k malým odchylkám v hodnotě kvality, protože čas čtení OPC je nejjistitelný a dochází tím k různě dlouhým časovým skokům čtené hodnoty otáček, proto není možné pro jedno nastavení modelu najít vhodné parametry pro všechny průběhy, lze však rozdíl kvalit rozdělit podle odchylek mezi skutečnou a simulovanou kvalitou.

Tab. 3.3-1: Porovnání Simulované a skutečné kvality

PM	C [kBaud]	T [s]	Kvalita simulovaného průběhu [%]	Kvalita skutečného průběhu [%]	d [%]
1,00	19,20	3,00	87,52	86,07	1,679
1,00	45,45	2,00	92,14	92,45	0,334
1,00	187,50	1,00	96,18	96,09	0,087
1,00	1500,00	2,00	99,06	99,11	0,051
2,00	19,20	2,00	79,03	86,12	8,231
2,00	45,45	2,00	90,99	92,71	1,860
2,00	187,50	3,00	98,53	97,91	0,629
2,00	1500,00	2,00	99,06	98,84	0,222
4,00	19,20	2,00	73,76	82,50	10,584
4,00	45,45	2,00	88,63	83,76	5,812
4,00	187,50	2,00	97,23	96,40	0,856
4,00	1500,00	5,00	99,62	98,79	0,843

Vzhledem k velké množině měřených dat ukazuje tabulka kvality pouze části vybraných průběhů od každé možnosti nastavení komunikace. Celkový přehled relativních odchylek kvality simulace a skutečného průběhu zobrazuje následující graf.



Graf 3.3-1: Histogram relativních odchylek Kvalit

Z uvedených hodnot je vidět větší relativní chyba při pomalé komunikaci a větším počtu FM. To je způsobeno větší citlivostí cyklu komunikace T_{Cykl} na žádost čtení dat OPC, kterými určuje průběh měření.

4. Porovnání různých metod řízení

Při porovnávání metod řízení se zde objevuje otázka kterou metodu použít, nelze zcela jasně říci která je nejvýhodnější. Toto rozhodnutí je závislé především na aplikaci, kde bude použito, nejen z hlediska kvality a rychlosti řízení, ale také z hlediska využití dané technologie a finančního, proto bych rád shrnul rozdíl mezi řídicími metodami které již byly analyzovány.

Jednou z těchto metod je řízení pomocí analogové veličiny, které se provádí pomocí D/A převodníku v PLC. Jednou z výhod tohoto řízení je poměrně rychlá aktualizace řídicí veličiny, protože každý řízený FM má svůj vlastní kanál a zápis se v programu na ně provádí okamžitě, pak je tedy rychlost řízení dána především rychlostí programu PLC a daných převodníků. Vzhledem k tomu že počet výstupů není neomezený, je v případě potřeby řídit více FM vybavit PLC rozšiřujícími moduly, což nemusí být zrovna levné. Dalším omezením je pouze přímovazební řízení bez možnosti měnit parametry a hlídat stav FM ze vzdáleného pracoviště, což může být při velké vzdálenosti zatěžující. Pro kvalitu dynamiky byl napsán vztah

$$[7, \text{s.38}] \quad \begin{aligned} K &= 100 && \dots \text{pro } f = 0 \text{ až } 0,1 \text{ Hz} \\ K &= -0,0823f^2 - 2,6851f + 103,91 && \dots \text{pro } f = 0,1 \text{ až } 20 \text{ Hz} \end{aligned}$$

Další metodou je řízení pomocí USS protokolu, jedná se o sběrnicové rozhraní typu Master-Slave. Výhodami u tohoto řízení je možnost připojit až 31 FM, které je možno vzdáleně nastavovat a hlídat jejich stav. Vzhledem k cyklické komunikaci s adresováním každého měniče dochází při zvýšení FM k lineárnímu nárůstu času, za který je daný měnič možné opět řídit a s tím související kvalitu dynamiky pro který byl napsán vztah [8, s.60]

$$\begin{aligned} Kvalita(T, T_{PX}, T_N) &= \frac{2 \cdot A \cdot \sin\left(\frac{p \cdot T_{PX}}{T}\right)}{n_{MAX}} \\ &\cdot T_N \left(\frac{0,35 \cdot T + 0,137 \cdot T \cdot T_{PX} - 0,216 + 233,34 \cdot T_{PX}}{T \cdot T_{PX}} \right) \\ &+ 99,86 - 0,062 \cdot T_{PX} - \frac{0,104 + 307,55 \cdot T_{PX}}{T} [\%] \end{aligned}$$

Oproti předchozímu řízení je podstatně pomalejší, což je v případě nutnosti potřeba kompenzovat vhodným nastavením FM.

Posledním způsobem řízení je pomocí sběrnice PROFIBUS, které je tématem této práce. Jedná se o sběrnicové rozhraní typu Master-Slave s možností připojení až

32 podřízených stanic. Tento způsob dává dohromady některé kladné vlastnosti předchozích způsobů, především počet připojitelných zařízení a vysokou rychlost řízení. Vzhledem k adresování každého měniče však také lineárně narůstá doba cyklu k možnosti aktualizace řízení FM, což lze kompenzovat možností nastavení komunikační rychlosti až 12Mbaud. Je však otázkou zda je takové vybavení potřeba k pomalým procesům.

5. Závěr

Diplomová práce se zabývá analýzou sběrnice PROFIBUS pro nalezení vztahu kvality dynamiky řízených pohonů v závislosti na konfiguraci pohonů a komunikace. Z daných výsledků lze říci že se jedná v závislosti na vysoké komunikační rychlosti o velmi rychlé spojení, které je však omezováno nejen parametry komunikace ale také rozsáhlostí řídicího programu.

V začátku práce byly vytvořeny funkce pro programovatelný automat, které umožňují vzdálené řízení a měření zaznamenaných hodnot pomocí technologie OPC. Funkce byly řešeny obecně s ohledem na univerzální použití různých komunikačních technologií a na různou volbu použitého zařízení, které podporuje daný programovací jazyk. Pomocí vývojového prostředí byla vytvořena aplikace využívající technologii OPC k řízení a vizualizaci měření PLC. Tyto oba vyvinuté programy jsou provázané mezi sebou a vzhledem k jejich nezávislosti na zařízení je možné je nezávisle použít po případných úpravách na danou aplikaci.

Vzhledem k neznalosti přesného popisu sběrnice byl analýzou proveden rozbor k určení komunikačních vlastností sběrnice a popis těchto vlastností dán do vztahů aby bylo možné použít tento rozbor obecně na základě měření. Tyto vztahy byli následně aplikovány v simulačním modelu řídicího systému s ohledem na již zmíněná omezení a byl tak vytvořen obecný model k simulování průběhů řízení a výpočet k určení kvality řízení při daných parametrech. Z hodnot kvalit při různém nastavení komunikace byl odvozen obecný vztah pro výpočet kvality, který je závislý pouze na poměru periody řízeného signálu a času cyklu řízení, což dává velký rozhled pro potřeby návrhu případných řídicích systému.

Celá práce byla zpracována tak, aby veškeré zjištěné poznatky byly použitelné obecně v různých aplikacích řídicích systému a práce tak mohla ulehčit rozhodování při případném návrhu již při obecném rozboru.

Seznam použité literatury

- [1] Siemens AG SIMATIC S7-300. Firemní dokumentace. 2004 A5E00105475-05
- [2] Siemens AG SIMATIC System Software. Firemní dokumentace. 2005 A5E00261410-02
- [3] Siemens AG SIMATIC Statement List (STL). Firemní dokumentace. 2004 A5E00261408-01
- [4] Siemens AG SIMATIC Communication with SIMATIC. Firemní dokumentace. 2003 C79000–G8976–C172
- [5] Siemens AG MICROMASTER 440 Parameter list. Firemní dokumentace. 2005 6SE6400-5BB00-0BP0
- [6] Siemens AG SIMODRIVE 611 universal/universal E. Firemní dokumentace. 2005 6SN1197–0AB20–1BP3
- [7] Košelja P. Analýza řízení otáček elektrických pohonů firmy Siemens ovládaných pomocí analogové veličiny. Diplomová práce, TUL 2004
- [8] Pirkl P. Analýza řízení otáček elektrických pohonů firmy Siemens ovládaného pomocí USS protokolu. Diplomová práce. TUL 2003

Seznam obrázků:

Obr. 1-1: Kompletní propojení zařízení sběrnicemi	11
Obr. 1.1.1-1: Protokol komunikace PROFIBUS-DP	14
Obr. 1.2-1: časování cyklu PLC	16
Obr. 1.3-1: Komunikace technologie OPC	17
Obr. 2.1.1-1: konfigurace v programu HW Config	20
Obr. 2.1.1-2: Propojení sběrnic mezi stanicemi v prog. NetPro	21
Obr. 2.1.8-1: vývojový diagram programu PLC	25
Obr. 2.2-1: Nastavení proměnných pro OPC server	26
Obr.2.2-2: Propojení kanálů OPC klient-server.....	27
Obr. 2.2.1-1: Panel Aplikace k řízení měření průběhů	28
Obr. 2.2.1-2: Panel aplikace k čtení a ukládání měření	29
Graf 3.1.1-1: Odečtení cyklu komunikace	31
Obr. 3.1.1-2: Rozdělení cyklu komunikace na úseky	31
Graf 3.1.2-1: zobrazení měření pro $C = 1,5$ MBaud	33
Graf 3.1.2-2: zobrazení měření pro $C = 187,5$ KBaud	34
Graf 3.1.2-3: zobrazení měření pro $C = 45,45$ KBaud	34
Graf 3.1.2-4: zobrazení měření pro $C = 19,2$ KBaud	35
Obr. 3.1.3-1: Model způsobu řízení	39
Graf 3.1.3-2: výřez průběhu k identifikaci	39
Graf 3.1.3-3: Výřez průběhu ke kontrole identifikace	40
Graf 3.1.3-4: Výřez průběhu ke kontrole identifikacec	40
Obr. 3.2-1: Určení plochy S_G	41
Obr. 3.2-2: Určení plochy S_R	42
Graf 3.2-3: Závislost Kvality naměřených průběhů ne periodě T	44
Graf 3.2-4: Plocha závislosti kvality na T a T_{Cykl}	45
Graf 3.2-5: Porovnání Kvality skutečného a simulovaného průběhu	45
Graf 3.3-1: Histogram relativních odchylek Kvalit	47

Seznam Tabulek

Tab. 1.1-1: Závislost maximální délky na přenosové rychlosti	11
Tab. 3.1.2-1: Zjištěné hodnoty cyklu komunikace T_{Cykl}	36
Tab. 3.1.2-2: Zjištěné hodnoty času komunikace s jedním FM T_{FM} [ms]	36
Tab. 3.1.2-3: časy obsluhy T_{cykl} [ms] daného počtu FM při dané C	37
Tab. 3.1.2-4: časy cyklu programu T_p [ms] pro dané nastavení komunikace	38
Tab. 3.2-1: Vypočtené Hodnoty kvality průběhů	43
Tab. 3.3-1: Porovnání Simulované a skutečné kvality	46

Příloha A – Seznam a popis napsaných funkcí

Seznam funkcí:

A.1	FC1 – SinGen.....	55
A.2	FC2 – RdFrDB.....	57
A.3	FC3 – TimeStmp.....	58
A.4	FC4 – WrToDB	62
A.5	FC5 – DB_Copy	63
A.6	FB1 – TotCreateDB	66
A.7	FC7 – TransDB.....	68
A.8	FC8 – ReqCopy	71

A.1 FC1 – SinGen

Popis funkce:

Funkce generuje sinusový průběh v závislosti na amplitudě, střední hodnotě a periodě požadovaného průběhu. Funkce má vlastní ovládání pro spouštění, nulování a povolení výstupu. Jednotky vstupních a výstupních parametrů pro hodnotu otáček jsou definovány jako procenta referenční frekvenci nastavené ve frekvenčním měniči ve vztahu výstupu 4000h =100% nastavené referenční frekvenci motoru

Použité proměnné:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
Ampl	Word	0.0	W#16#0	Amplituda zmeny otacek
Ostr	Word	2.0	W#16#0	Stredni hodnota otacek
Period	DWord	4.0	DW#16#0	delka periody v ms
CycleTime	Int	8.0	0	delka minuleho cyklu PLC
Run	Bool	10.0	FALSE	bit spousteni
Reset	Bool	10.1	FALSE	bit nulovani casu
Enabled	Bool	10.2	FALSE	Povoleni zmeny hodnot otacek
OUT		0.0		
IN_OUT		0.0		
eltime	DWord	12.0	DW#16#0	Ubehly cas v spusteni
Otacky	Word	16.0	W#16#0	Vystupni hodnota otacek
TEMP		0.0		
RETURN		0.0		
RET_VAL		0.0		

Ampl – vstupní Word udávající amplitudu generovaného signálu, pro změnu počáteční fáze může nabývat záporných hodnot.

Ostr – vstupní Word udávající hodnotu středních otáček generovaného signálu, může nabývat záporných hodnot.

Period – vstupní Doubleword udávající dobu periody generovaného signálu v ms.

CycleTime – čas předešlého cyklu v ms, hodnota je dána systémovým parametrem v PLC,tato proměnná je v každém cyklu přičítána k uběhlému času eltime.

Run – vstupní Bit spouštění generátoru, signál je aktivní v log.,,1“, po přechodu do log.,,0“ je generátor pozastaven a na výstupu zůstává poslední vypočítaná hodnota otáček.

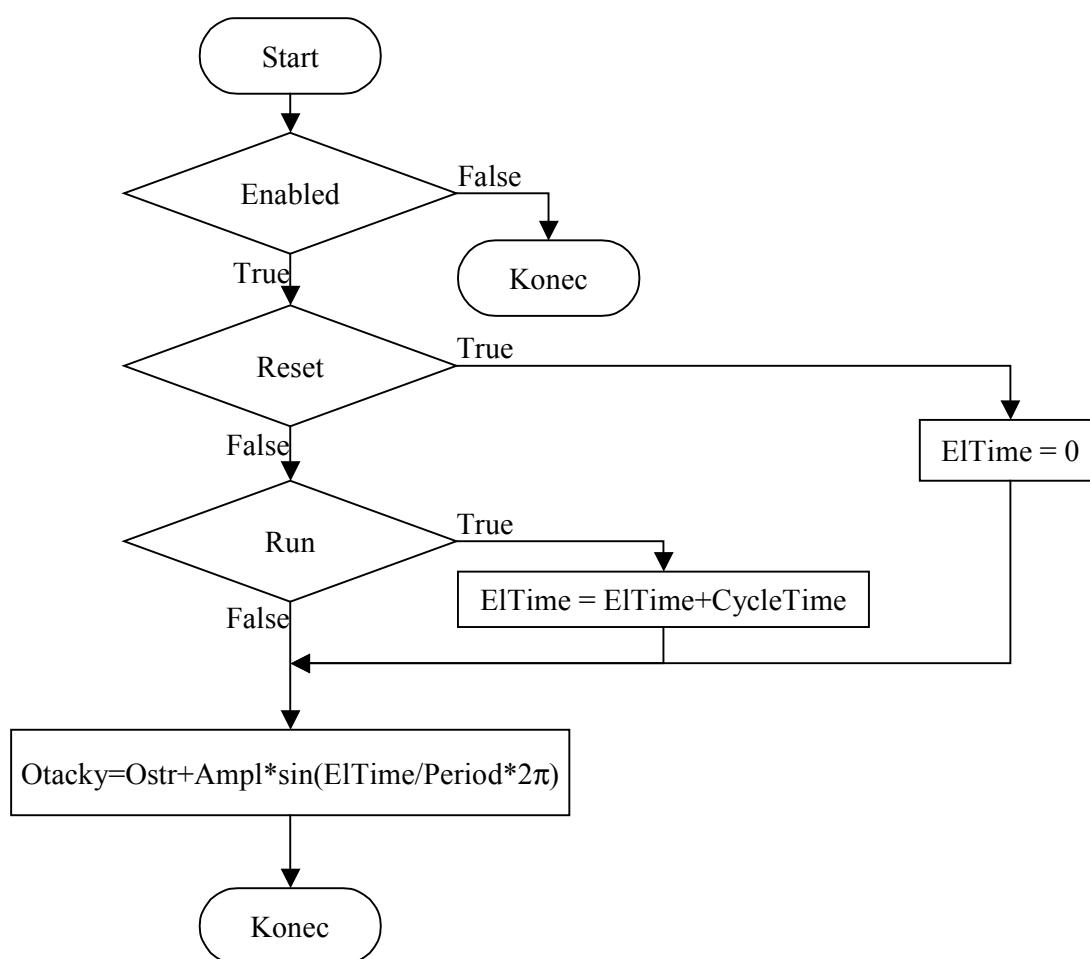
Reset – vstupní Bit nulování uběhlého času, signál je aktivní v log.,,1“, čas je nastaven na 0 a uložen, výstup je roven Ostr.

Enabled – vstupní Bit povolení výstupu generátoru, signál je aktivní v log.,1“, v log.,0“ není na výstup proveden zápis generátorem, hodnota zůstává stejná jako před voláním této funkce.

eltime – vstupně-výstupní Doubleword uběhlého času, před výpočtem výstupních otáček je za běhu generátoru přičten čas předchozího cyklu PLC.

Otacky – vstupně-výstupní Word výstupu vypočtené hodnoty otáček, hodnota je vypočítána při povoleném výstupu dle vztahu $Otacky = Ostr + Ampl \times \sin\left(\frac{eltime}{Period} \times 2\pi\right)$

Algoritmus funkce:



A.2 FC2 – RdFrDB

Popis funkce:

Funkce čte jeden Word z pozice dané parametrem Incr začínající na pozici 0, z DB bloku, ve vnitřní paměti PLC, daného číslem parametru NoDB a následně ho zapíše na pevně stanovenou adresu ODB.

Použité proměnné:

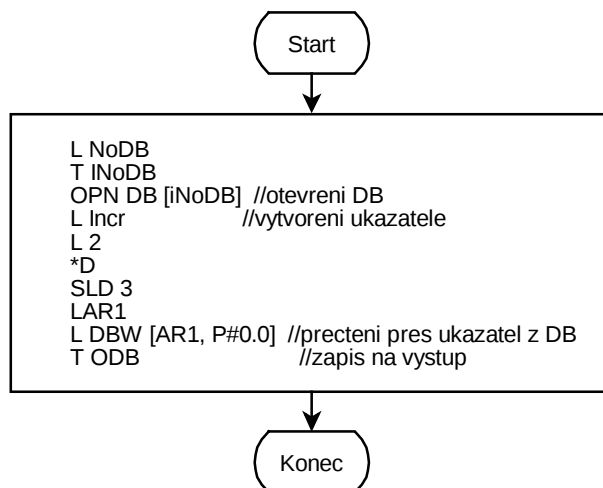
Name	Data Type	Address	Initial Value	Comment
IN		0.0		
NoDB	Word	0.0	W#16#0	Cislo ctene DB
Incr	Word	2.0	W#16#0	Cislo cteneho Wordu od 0
OUT		4.0		
ODB	Word	4.0	W#16#0	Zapisovany Word
IN_OUT		0.0		
TEMP		0.0		
INoDB	Word	0.0		Pomocny Word pro syntax prikazu
RETURN		0.0		
RET_VAL		0.0		

NoDB – vstupní Word čísla čteného DB bloku, je nutno ohlídat chybové stavy otevírání neexistujících DB.

Incr – vstupní Word hodnoty pozice čteného wordu začínající od pozice 0, parametr je vnitřně ve funkci násoben 2, aby byla zjištěna adresa prvního bytu. Je nutno ohlídat chybové stavy překročení délky čteného DB bloku.

ODB – výstupní Word přečtené hodnoty.

Algoritmus funkce:



A.3 FC3 – TimeStmp

Popis funkce:

Funkce podle nastavení parametrů mění ukazatel na pozici wordu v bloku DB, toho je využito při záznamu hodnot měření. Může pracovat ve třech nastavitelných režimech, jejichž popis je seřazen v sestupné prioritě:

Auto – při každém n-tém cyklu, nastaveném v TimeDiv zvětší ukazatel o 1.

Hand – ukazatel je zvětšen o 1 při sestupné hraně Trg

Time – funkce čte hodnotu času z DB bloku s číslem NoDB a ukazatelem Incr, tuto hodnotu pak násobí hodnotou TimeDiv. Překročí-li automaticky přičítaný čas ElTime vypočítanou hodnotu, zvýší ukazatele o 1.

Použité proměnné:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
NoDB	Word	0.0	W#16#0	cislo DB
MaxIncr	Word	2.0	W#16#0	maximalni pozice na radku v DB
LaTime	Int	4.0	0	cas minuleho cyklu
ResetT	Bool	6.0	FALSE	1-nulovani casu
ResetIn	Bool	6.1	FALSE	1-nulovani ukazatele
Run	Bool	6.2	FALSE	1-spusteno citani
Hnd_Tim	Bool	6.3	FALSE	0-Timer 1-rucni spinani
Trg	Bool	6.4	FALSE	ovladani aktivni na sestupnou hranu
TimeDiv	DWord	8.0	DW#16#0	Nasobek casu
Auto	Bool	12.0	FALSE	1-Automaticky Increment
OUT		14.0		
OverFlow	Bool	14.0	FALSE	1-Presazeni maximalni hodnoty Incr
IN_OUT		16.0		
Incr	Word	16.0	W#16#0	ukazatel radky v DB
ElTime	DWord	18.0	DW#16#0	cas ubehly od 1. spusteni
CIFlag	Bool	22.0	FALSE	minuly stav TRG
TEMP		0.0		
INoDB	Word	0.0		

NoDB – vstupní Word čísla DB bloku ze kterého jsou čteny časové hodnoty v režimu Time.

MaxIncr – vstupní Word maximální hodnoty ukazatele Incr, po pokusu zvýšit ukazatele nad tuto mez zůstává hodnota Incr nezměněna a je nastaven výstup OverFlow.

LaTime – vstupní Word času předešlého cyklu v ms, hodnota je dána systémovým parametrem v PLC, tato proměnná je v každém cyklu přičítána k uběhlému času eltime.

ResetT – vstupní Bit nulování uběhlého času ElTime aktivního v log. „1“.

ResetIn – vstupní Bit nulování ukazatele Incr aktivního v log. „1“.

Run – vstupní Bit spouštění funkce aktivní v log. „1“, je-li neaktivní, je pouze ukládán stav Trg a může být aktivováno nulování. Tento vstup ovlivňuje všechny režimy chodu.

Hnd_Tim – vstupní Bit pro nastavení režimu Hand (log. „1“) nebo Time (log. „0“), pokud je nastaven režim Auto, nemá tento bit žádnou funkci.

Trg – vstupní Bit pro ovládání v režimu Hand, při detekci sestupné hrany se ukazatel zvýší o 1.

TimeDiv – vstupní Doubleword pro nastavení chodu režimu, v režimu Auto nastavuje po kolika cyklech programu se zvýší ukazatel, v režimu Time nastavuje násobek časové jednotky, v režimu Hand není využit.

Auto – vstupní Bit pro nastavení režimu Auto, aktivní v log. „1“.

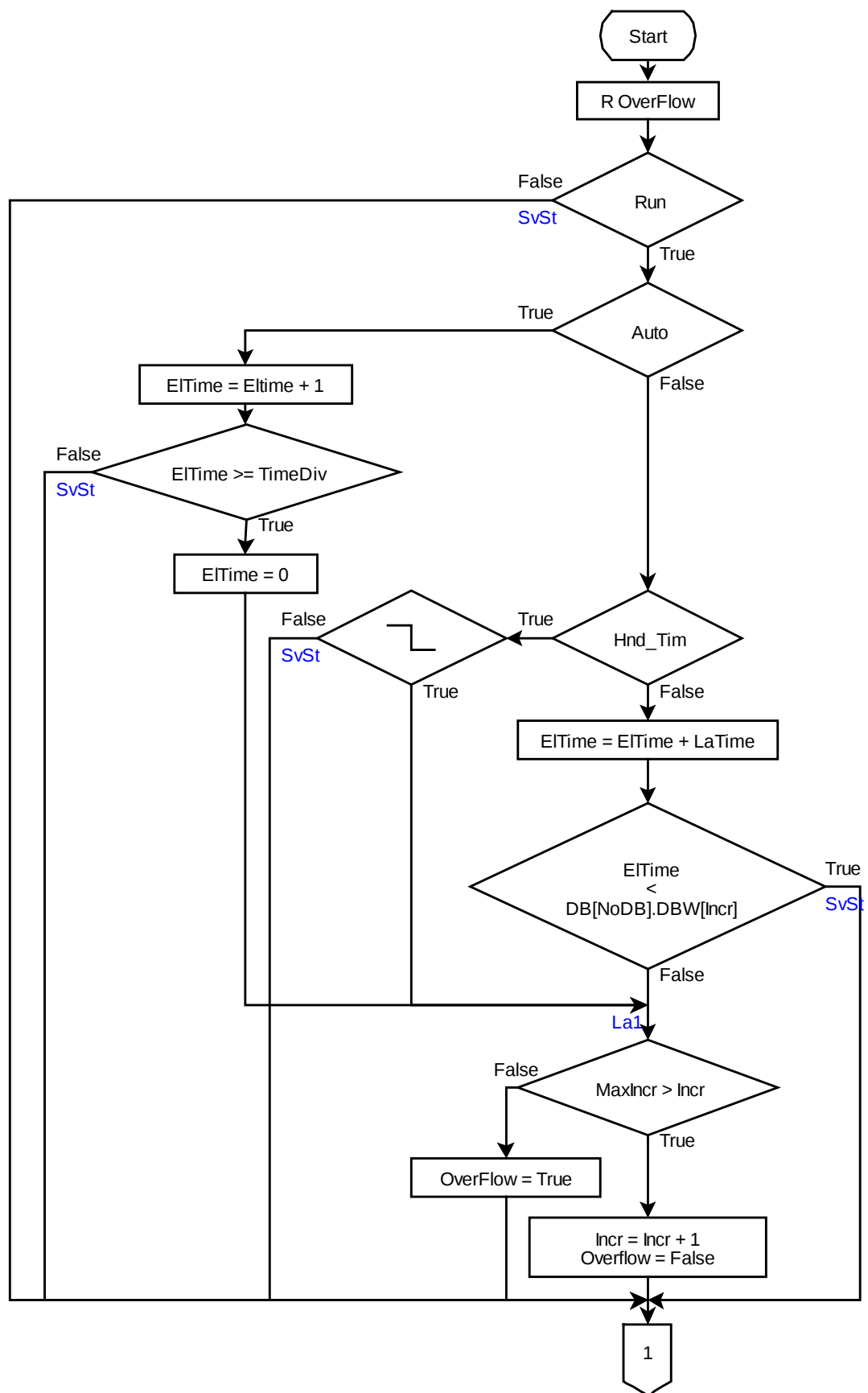
Overflow – výstupní Bit signalizující stavem log. „1“ pokus o překročení hodnoty MaxIncr, signalizace je použita ve všech režimech.

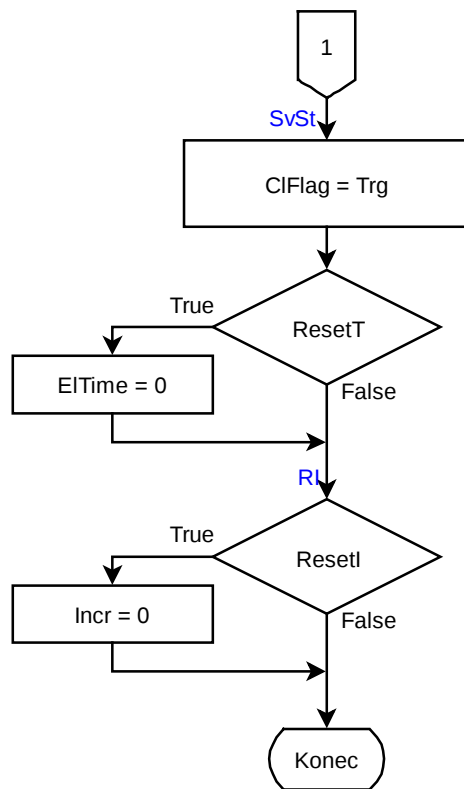
Incr – vstupně-výstupní Word ukazatele na pozici v DB bloku.

ElTime – vstupně-výstupní Doubleword, v režimu Time jde o záznam hodnoty uběhlého času, v režimu Auto jde o záznam počtu uběhlých cyklu po posledním zvýšení ukazatele, při zastavení běhu bitem Run není hodnota měněna.

CIFlag – vstupně-výstupní Bit k uchování posledního stavu Trg pro detekci sestupné hrany v režimu Hand.

Algoritmus funkce:





A.4 FC4 – WrToDB

Popis funkce:

Funkce zapisuje jeden Word z pevně dané adresy IDB do DB bloku, ve vnitřní paměti PLC, daného číslem parametru NoDB na pozici dané parametrem Incr začínající na pozici 0.

Použité proměnné:

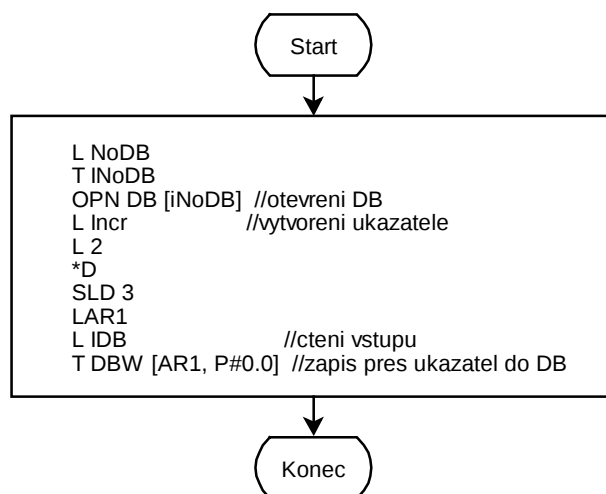
Name	Data Type	Address	Initial Value	Comment
IN		0.0		
NoDB	Word	0.0	W#16#0	Cislo zapisovane DB
Incr	Word	2.0	W#16#0	Cislo zapisovaneho Wordu od 0
IDB	Word	4.0	W#16#0	Cteny Word
OUT		0.0		
IN_OUT		0.0		
TEMP		0.0		
INoDB	Word	0.0		Pomocny word pro syntax prikazu
RETURN		0.0		
RET_VAL		0.0		

NoDB – vstupní Word čísla zapisovaného DB bloku, je nutno ohlídat chybové stavy otevírání neexistujících DB.

Incr – vstupní Word hodnoty pozice zapisovaného wordu začínající od pozice 0, parametr je vnitřně ve funkci násoben 2, aby byla zjištěna adresa prvního bytu. Je nutno ohlídat chybové stavy překročení délky čteného DB bloku.

ODB – vstupní Word zapisované hodnoty.

Algoritmus funkce:



A.5 FC5 – DB_Copy

Popis funkce:

Funkce kopíruje data mezi DB bloky, které mohou být v pracovní paměti PLC nebo na MMC. Funkce využívá a zapouzdřuje systémové funkce (SFC), SFC 20 „BLKMOV“ pro kopírování mezi DB bloky v pracovní paměti PLC, SFC 83 „READ_DBL“ pro kopírování DB bloku z MMC do pracovní paměti, SFC 84 „WRIT_DBL“ pro kopírování DB bloku z pracovní paměti do MMC, tak, aby bylo možné nastavením parametrů vybrat danou funkci a vytvořit hodnoty pro její parametry a vyhnout se nepřehlednosti v hlavním programu. Pro přehlednost a vysvětlení vývojového diagramu budou popsány i dočasné vnitřní proměnné.

Použité proměnné:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
N_DBSrc	Word	0.0	W#16#0	Cislo DB zdroje
I_DBSrc	Word	2.0	W#16#0	Zacatek v DB zdroje
N_DBDst	Word	4.0	W#16#0	Cislo DB cile
I_DBDst	Word	6.0	W#16#0	Zacatek v DB Cile
BPR	Word	8.0	W#16#0	Pocet bytu na zaznam
WMMC	Bool	10.0	FALSE	1-kopirovani mezi MMC 0- ve Work
To_FrMMC	Bool	10.1	FALSE	1-do MMC 0-z MMC
OUT		12.0		
IN_OUT		12.0		
Cnt_Rec	Word	12.0	W#16#0	Pocet kop.zaznamu
RetVal_IO	Int	14.0	0	Navratova hodnota
TEMP		0.0		
PSrc	Any	0.0		Adresa oblasti zdroje
PDst	Any	10.0		Adresa oblasti cile
Req	Bool	20.0		pozadavek spusteni SFC
TmpRetVal	Int	22.0		Docasna navrat.hodn. od SFC
TmpBusy	Bool	24.0		SFC zaneprazdnena

N_DBSrc – vstupní Word čísla zdrojového DB bloku, je třeba ohlídat existenci a umístění v nastavené paměti.

I_DBSrc – vstupní Word čísla pozice prvního záznamu ve zdrojovém DB bloku pro kopírování, číslování záznamů v DB je od 0.

N_DBDst – vstupní Word čísla cílového DB bloku, je třeba ohlídat existenci a umístění v nastavené paměti.

I_DBDst – vstupní Word čísla pozice prvního záznamu v cílovém DB bloku pro kopírování, číslování záznamů v DB je od 0.

BPR – vstupní Word počtu bytů na záznam, tímto parametrem je přepočítávána oblast pro kopírování z počtu záznamů na počet bytů.

WMMC – vstupní Bit pro nastavení kopírování, pro hodnotu log. „1“ nastavuje kopírování mezi MMC a paměti PLC, pro log. „0“ nastavuje kopírování v paměti PLC.

To_FrMMC – vstupní Bit pro nastavení směru kopírování při nastavení WMMC=log. „1“, pro hodnotu log. „1“ nastavuje kopírování z paměti PLC do MMC, pro hodnotu log. „0“ nastavuje kopírování z MMC do paměti PLC. Při nastavení WMMC= log. „0“ nemá využití.

Cnt_Rec – vstupně-výstupní Word počtu kopírovaných záznamů, funkce po kompletním dokončení kopírování nuluje tuto hodnotu aby nedošlo k opakovanému kopírování.

RetValIO – vstupně-výstupní Word návratové hodnoty hlášení průběhu funkcí SFC83 a SFC84, které aktivují asynchronní kopírování s MMC.V dalším cyklu programu je možné zjistit stav vyvolaného kopírování, po dokončení je hodnota nulována.

PSrc – vnitřní proměnná pro uložení vypočteného ukazatele na oblast zdroje kopírování, která je vstupním parametrem pro volané SFC.

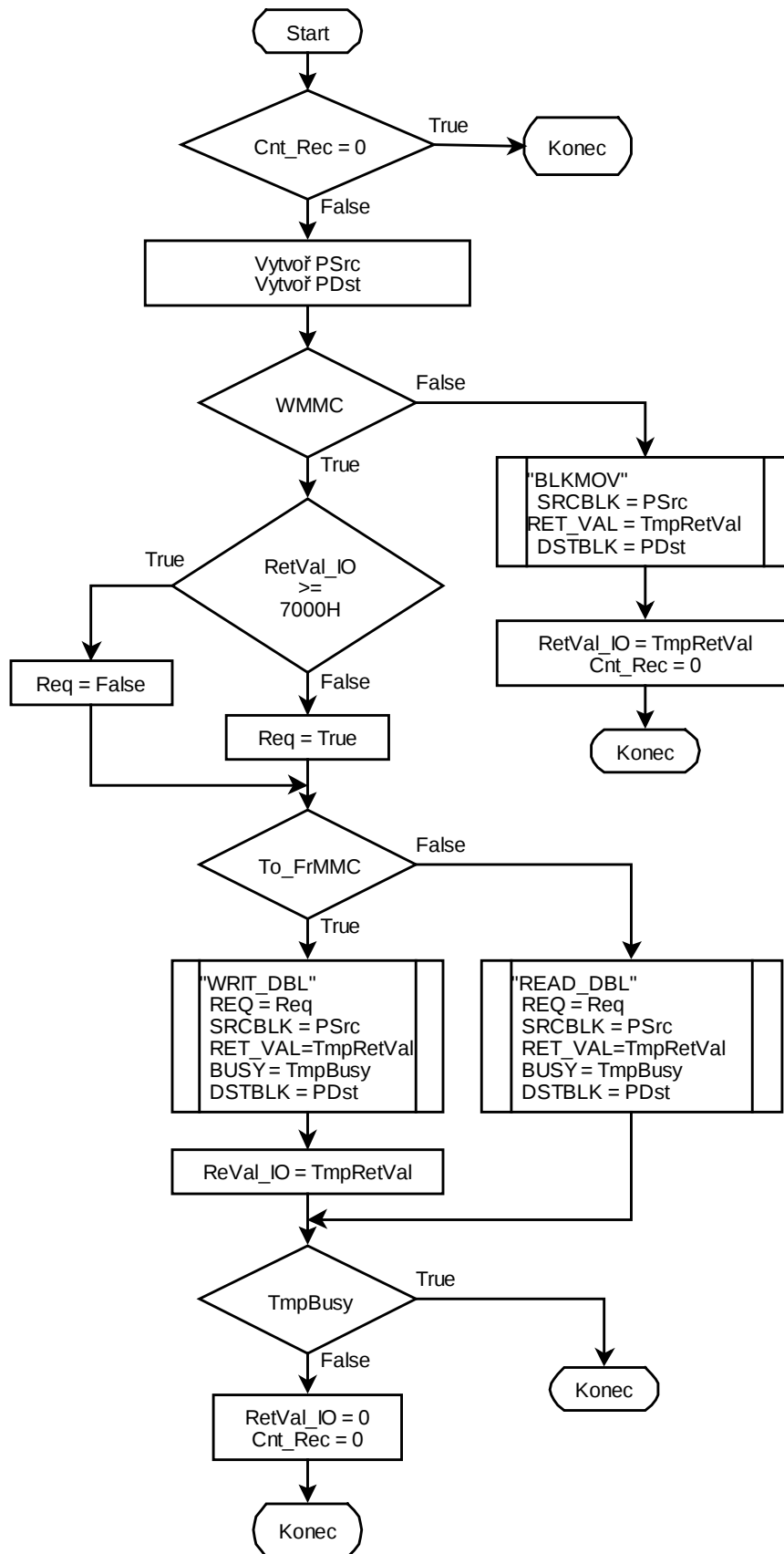
PDst – vnitřní proměnná pro uložení vypočteného ukazatele na oblast cíle kopírování, která je vstupním parametrem pro volané SFC,.

Req – vnitřní proměnná k nastavení požadavku kopírování, je-li hodnota log. „0“, funkce pouze vrací stav kopírování.

TmpRetVal – vnitřní proměnná pro uložení návratové hodnoty chybového hlášení funkce.

TmpBusy – vnitřní proměnná k uložení stavu kopírování, v log. „1“ kopírování stále probíhá, pro log. „0“ je kopírování dokončeno.

Algoritmus funkce:



A.6 FB1 – TotCreateDB

Popis funkce:

Funkce dynamicky vytváří DB bloky s danou velikostí a číslem, blok může být vytvořen podle nastavení ve vnitřní paměti nebo v MMC, pokud blok existuje a má jinou délku než požadovanou, je smazán a vytvořen nový s žádanou délkou. Funkce využívá systémové funkce SFC 24 „TEST_DB“ k zjištění informací o DB bloku, SFC 23 „DEL_DB“ ke smazání DB bloku, SFC 22 „CREAT_DB“ k vytvoření DB bloku ve vnitřní paměti PLC a SFC 82 „CREA_DBL“ k vytvoření DB bloku na MMC.

Použité proměnné:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
CntByte	Word	0.0	W#16#0	Pocet Bytu v DB
NumDB	Word	2.0	W#16#0	Cislo DB
WMMC	Bool	4.0	FALSE	WMMC=1 DB v MMC
OUT		6.0		
IN_OUT		6.0		
RetValIO	Int	6.0	0	Navratova hodnota
STAT		8.0		
SrcBlk	Word	8.0	W#16#0	Pro syntax "CREA_DBL"
TEMP		0.0		
Ret_TestDB	Int	0.0		
DBLength	Word	2.0		

CntByte – vstupní Word počtu bytů v DB bloku, musí být násobkem čísla 2.

NumDB – vstupní Word čísla vytvořeného DB bloku, nutno zajistit neexistenci DB se stejným číslem, jinak by byla přepsána.

WMMC – vstupní Bit nastavení volby výběru oblasti paměti vytvoření, pro log. „1“ je výběr MMC, pro log. „0“ vnitřní paměť PLC.

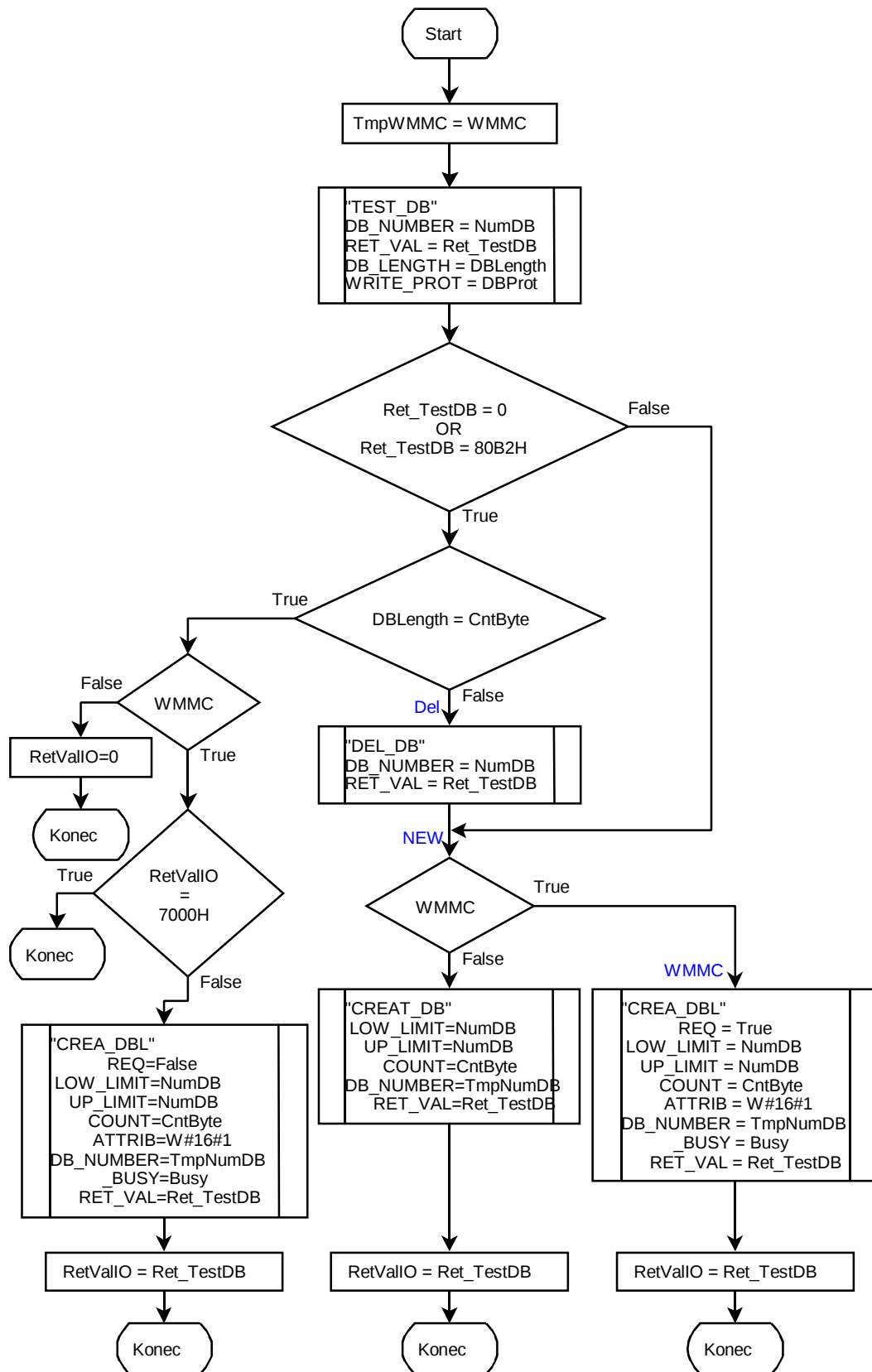
RetValIO – vstupně-výstupní Word k uchování chybového hlášení ukončení SFC.

SrcBlk – vnitřní statická proměnná pro správnou syntaxi volání funkce SFC 82 „CREA_DBL“, hodnotou, kterou bude vyplněna, bude naplněn vytvořený DB blok.

Ret_TestDB – vnitřní proměnná k uchování stavu v jakém je testovaný DB blok.

DBLength – vnitřní proměnná k uchování délky testovaného DB bloku, je-li rozdílná od požadované délky.

Algoritmus funkce:



A.7 FC7 – TransDB

Popis funkce:

Funkce dle nastavení kopíruje data ze vstupního DB bloku na výstupní DB blok s nastavenou změnou pořadí wordů, nastavení umožňuje přesunout daný word vpřed i vzad z původní pozice, nebo daný word vůbec nezapsat. Funkce má omezení v počtu přesunů maximálně na osm včetně vynechaných wordů a omezení v rozsahu pozic přesunů na maximálně patnáctou pozici od začátku DB bloku. Funkce využívá funkce FC2 „RdFrDB“ pro postupné čtení wordů ze vstupního DB bloku a FC4 „WrToDB“ pro zápis na danou pozici do výstupního DB bloku, dále jsou použity příkazy SRD (resp. SLD) pro posun bitů o daný počet ve wordu vpravo (resp. vlevo).

Popis parametrů:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
N_DBIn	Word	0.0	W#16#0	Cislo vstupni DB
N_DBOut	Word	2.0	W#16#0	Cislo vystupni DB
CoTr	DWord	4.0	DW#16#0	Poradnice Wordu
N_Words	Word	8.0	W#16#0	Pocet Wordu k presunu
OUT		10.0		
IN OUT		10.0		
TEMP		0.0		
CntTr	Word	0.0		index vstupniho WORDu
TmpW	Word	2.0		Hodnota vstupniho Wordu
TmpTrans	Word	4.0		Index Vystupniho wordu
Shift	Word	6.0		posun CoTr
TmpCoTr	DWord	8.0		Docasne poradnice pro vypocet pozice

N_DBIn – vstupní Word čísla zdrojového DB bloku, musí být vytvořen ve vnitřní paměti PLC.

N_DBOut – vstupní Word čísla cílového DB bloku, musí být vytvořen ve vnitřní paměti PLC.

CoTr – vstupní Doubleword pořadnic kopírovaných wordů zaznamenaných v hexadecimální číselné soustavě, kde číslo nejvyššího řádu, který je použit k přesunu nastavením parametru N_Words, značí pozici, začínající od 1, na kterou bude zapsán první word zdrojového bloku. Je-li hodnota hexadecimální číslice 0, pak není daný word zapisován do cílového bloku.

N_Words – vstupní Word počtu přesouváných wordů od počátku zdrojového DB bloku, včetně na začátku nezapisovaných. tento parametr se nastavuje na počet řádů

použitých číslíc v parametru CoTr, případně se přičítá počet wordů od začátku bloku, které nemají být zapsány.

CntTr – vnitřní proměnná pro určení pozice čtení ze zdrojového DB bloku.

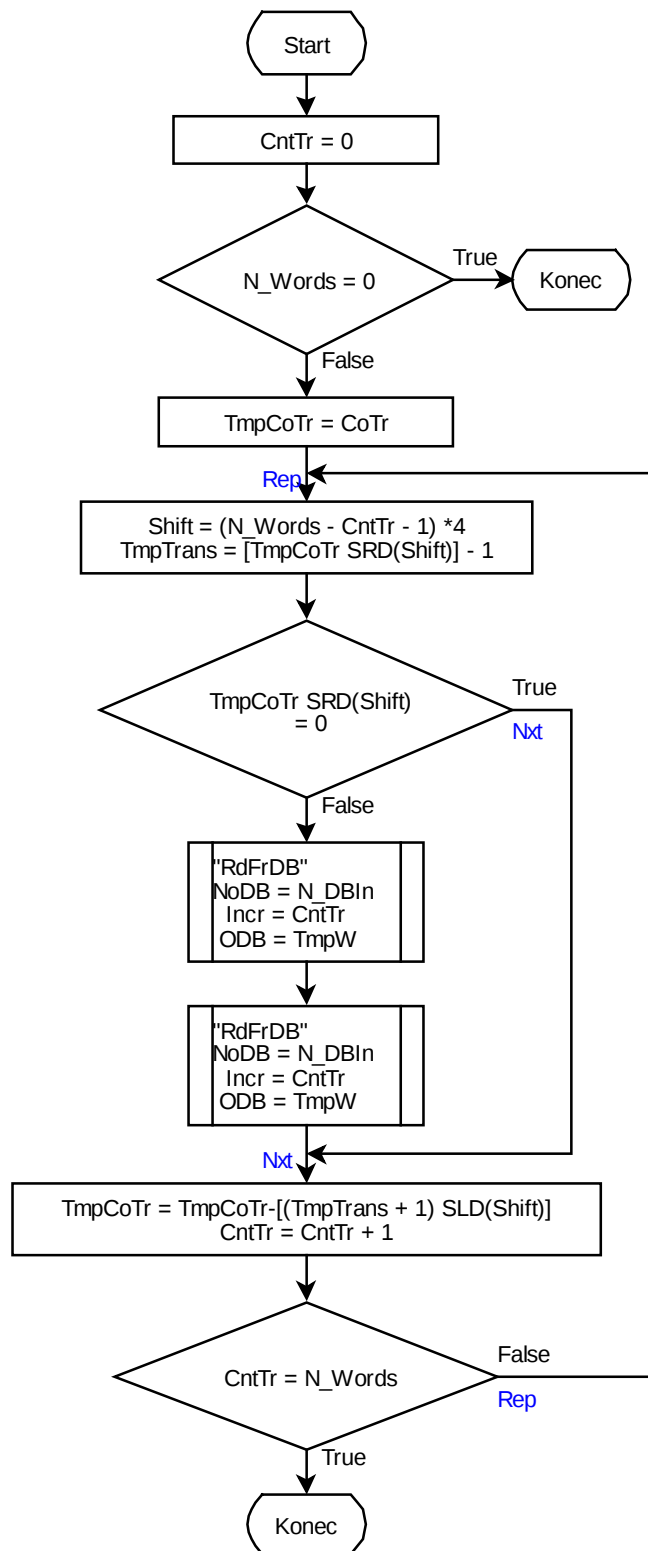
TmpW – vnitřní proměnná pro dočasné uložení hodnoty čteného wordu.

TmpTrans – vnitřní proměnná pro určení pozice zápisu do cílového DB bloku.

Shift – vnitřní proměnná pro uložení výpočtu hodnoty posunu bitů v TmpCoTr.

TmpCoTr – vnitřní proměnná pro postupné výpočty pořadnic CoTr.

Algoritmus funkce:



A.8 FC8 – ReqCopy

Popis funkce:

Funkce je navržena především pro kopírování měřených a žádaných hodnot mezi vnitřní pamětí a MMC.

Funkce automaticky kopíruje data mezi vnitřní pamětí a MMC v závislosti na inkrementu funkce „TimeStmp“ tak, že DB blok ve vnitřní paměti je rozdělen na polovinu a je-li inkrement zvýšen nad tuto mez, je kopírována nižší polovina, dojde-li inkrement na konec DB bloku, je nulován a kopíruje se vyšší polovina DB bloku.

Při kopírování dochází ke stránkování DB bloku na MMC, přičemž zápis na MMC probíhá na aktuální stránce, ale čtení z MMC probíhá na stránce následující aby byla dodržena kontinuita měření.

Při tomto kopírování je použita funkce „DB_Copy“, vzhledem k tomu, že jsou funkce pro kopírování asynchronní k běhu PLC, je tímto kopírováním zatěžován běh PLC jen nejnútnejšími výpočty pro nastavení kopírování.

Popis parametrů:

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
MaxIncr	Word	0.0	W#16#0	Maximalni pozice AbsIncr
MaxWork	Word	2.0	W#16#0	Maximalni pocet Wordu v prac.DB
OverFlow	Bool	4.0	FALSE	Bit dosazeni max hodnoty ukazatele
WrSrc	Word	6.0	W#16#0	Cislo 1.DB - zdroje zapisu do MMC
WrDes	Word	8.0	W#16#0	Cislo 1.DB - cile zapisu do MMC
WrCnt	Word	10.0	W#16#0	Pocet DB pro zapis
RdSrc	Word	12.0	W#16#0	Cislo 1.DB - zdroje cteni z MMC
RdDes	Word	14.0	W#16#0	Cislo 1.DB - cile cteni z MMC
RdCnt	Word	16.0	W#16#0	Pocet DB pro cteni
OUT		0.0		
IN_OUT		18.0		
ReqCopy	Bool	18.0	FALSE	Pozadavek dalsiho kopirovani
RelIncr	Word	20.0	W#16#0	Hodnota relativni pozice ukazatele
AbsIncr	Word	22.0		Posun relativniho ukazatele
ActDB	Word	24.0	W#16#0	Cislo aktualne kopirovane DB
ActWrRd	Bool	26.0	FALSE	Stav funkce 1-zapis 0-cteni
RetValIO	Int	28.0	0	Navratova hodnota kopirovani
HiLoDB	Bool	30.0	FALSE	Posledni kopirovani 1-vyssi 0-nizssi
AutoRun	Bool	30.1	FALSE	Stav spusteni funkci
RelIncrMax	Word	32.0	W#16#0	Maximalni hodnota relativni pozice ukazatele

MaxIncr – vstupní Word maximální hodnoty absolutní pozice wordu v DB bloku v MMC, slouží k omezení kopírování na konci DB bloku tak, aby nedošlo k překročení rozsahu za konec DB bloku.

MaxWork – vstupní Word počtu wordů v DB bloku, umístěném ve vnitřní paměti PLC. Výpočtem z parametru se vypočítává polovina v DB bloku k určení meze kopírování na dolní a horní polovinu.

Overflow – vstupní Bit indikace dosažení konce DB bloku ukazatelem pozice RelIncr, po nastavení do log. „1“ se aktivuje kopírování v horní polovině DB bloku ve vnitřní paměti s MMC.

WrSrc – vstupní Word čísla prvního zdrojového DB bloku, ve vnitřní paměti, od kterého začíná kopírování do prvního cílového DB bloku v MMC. Je třeba zajistit umístění DB bloků ve vnitřní paměti a jejich souvislou řadu.

WrDes – vstupní Word čísla prvního cílového DB bloku v MMC, do kterého je kopírován první DB blok z vnitřní paměti. Je třeba zajistit umístění DB bloků v MMC a jejich souvislou řadu.

WrCnt – vstupní Word hodnoty počtu kopírovaných DB bloků z vnitřní paměti do MMC. Je-li hodnota rovna nule, nedojde ke kopírování.

RdSrc – vstupní Word čísla prvního zdrojového DB bloku, v MMC, od kterého začíná kopírování do prvního cílového DB bloku ve vnitřní paměti. Je třeba zajistit umístění DB bloků v MMC a jejich souvislou řadu.

RdDes – vstupní Word čísla prvního cílového DB bloku, ve vnitřní paměti, do kterého je kopírován první DB blok z MMC. Je třeba zajistit umístění DB bloků ve vnitřní paměti a jejich souvislou řadu.

RdCnt – vstupní Word hodnoty počtu kopírovaných DB bloků z MMC do vnitřní paměti. Je-li hodnota rovna nule, nedojde ke kopírování.

ReqCopy – vstupně-výstupní Bit, který si funkce nastavuje sama při začátku kopírování dané části DB bloků do log. „1“, zpět do log. „0“ si jej nastavuje po ukončení kopírování dané části DB bloků. Tento bit by neměl být ovlivňován jinak, než samotnou funkcí, a při startu PLC by měl být nastaven do log. „0“

RelIncr – vstupně-výstupní Word hodnoty ukazatele na aktuální pozici wordu v DB bloku, podle tohoto parametru je spouštěno kopírování pro dolní polovinu DB bloku při překročení poloviny maximální délky MaxWork. Při aktivaci kopírování parametrem Overflow, kdy dojde k dosažení maximální hodnoty ukazatele, je RelIncr nulován.

AbsIncr – vstupně-výstupní Word hodnoty posunu relativního ukazatele na pozici wordu v DB bloku, parametr udává začátek aktuální části DB bloku v MMC, do které se budou zapisovat DB bloky z vnitřní paměti, pro čtení případných

žádaných hodnot z MMC je hodnota parametru vnitřně zvýšena o MaxWork, aby byly do DB bloku, ve vnitřní paměti, zapsány následující žádané hodnoty. Parametr je při aktivování kopírování parametrem Overflow zvýšen funkcí o hodnotu MaxWork a neměl by být během činnosti programu měněn vnějším zásahem.

ActDB – vstupně-výstupní Word čísla právě kopírovaného zdrojového DB bloku, pomocí kterého funkce zjišťuje stav kopírování. Parametr je funkcí sám automaticky nastavován při spuštění kopírování a neměl by být měněn.

ActWrRd – vstupně-výstupní Bit vnitřně nastavovaný funkcí k určení, zda je funkce ve stavu zápisu, kdy je bit v log. „1“, nebo čtení, kdy je bit v log. „0“, vůči MMC. Podle hodnoty nastavuje používanou funkci „DB_Copy“ do správného směru kopírování.

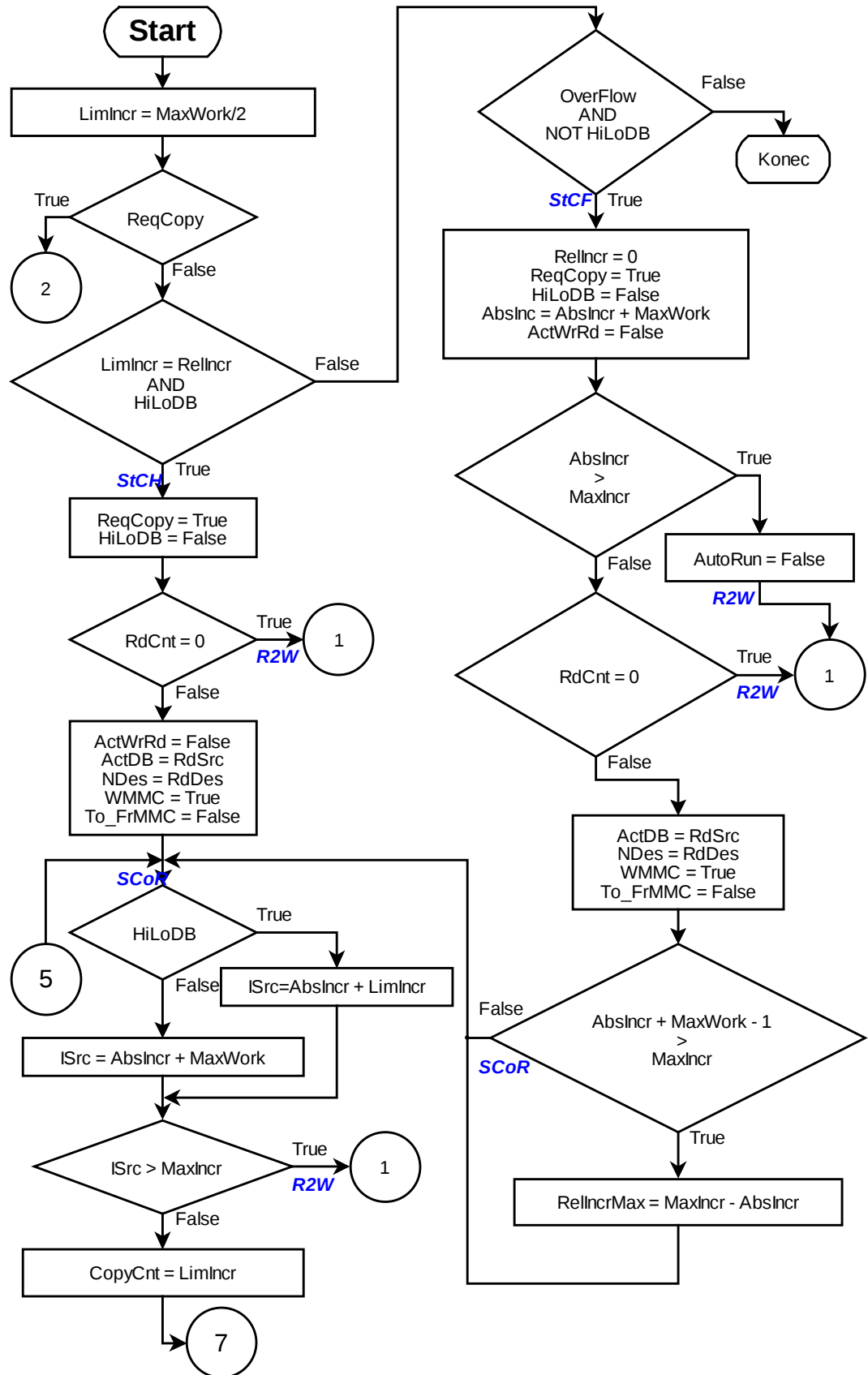
RetValIO – vstupně-výstupní Word k uložení stavu při ukončení funkce a následném zjištění stavu při dalším volání funkce v dalším cyklu programu.

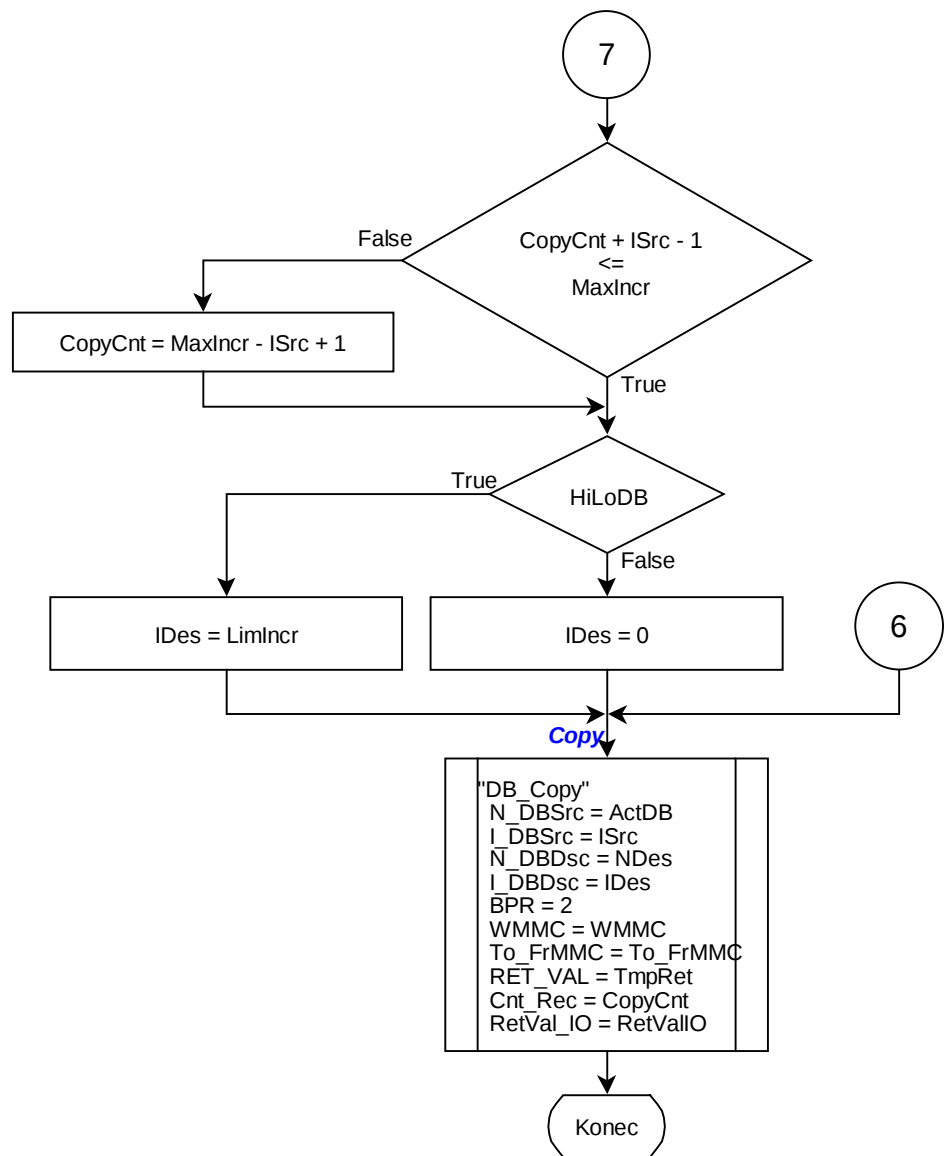
HiLoDB – vstupně-výstupní Bit k uložení stavu, v jaké polovině DB bloku je aktivované kopírování (log. „1“ pro horní polovinu, log. „0“ pro dolní polovinu), podle kterého dochází k výpočtům mezi kopírování. Při inicializaci PLC by měl být nastaven do log. „1“, a za běhu programu neměl být měněn.

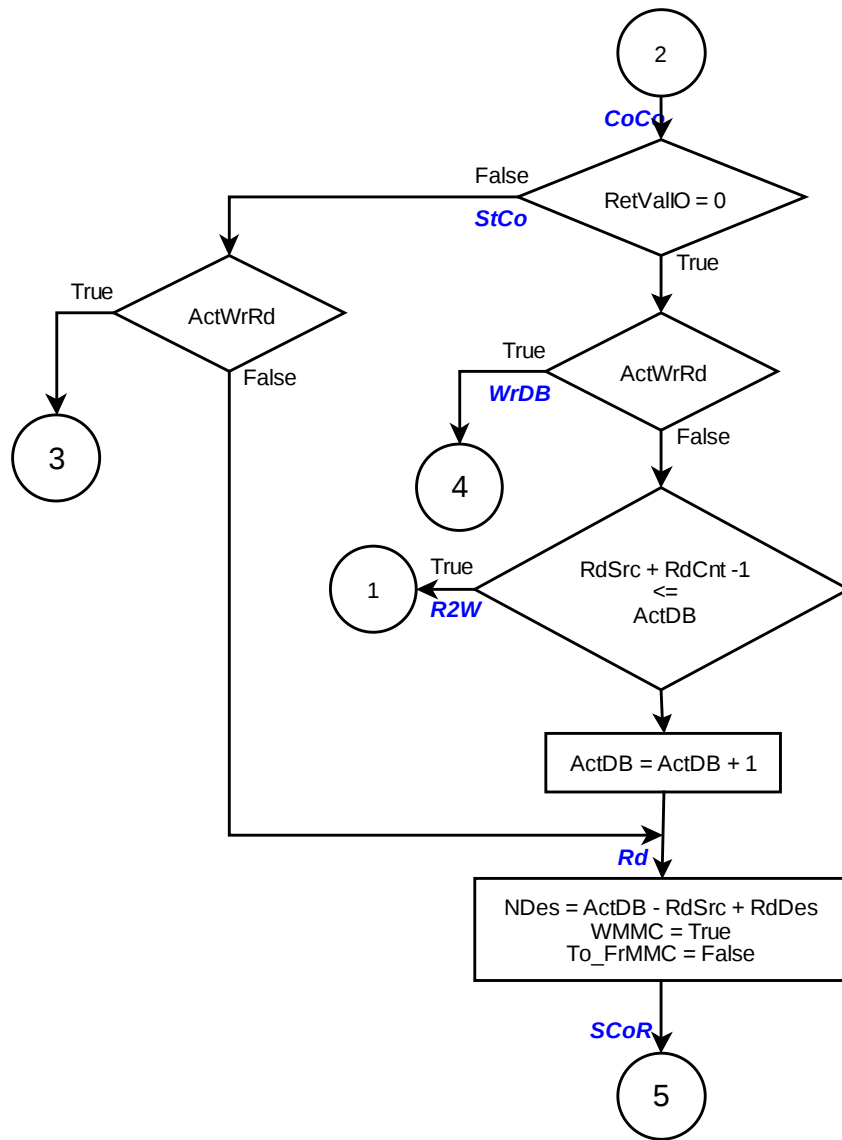
AutoRun – vstupně-výstupní Bit k signalizaci dosažení konce DB bloku v MMC, při kopírování, nastavením do log. „0“, jinak bit zůstává nezměněn od vstupního stavu.

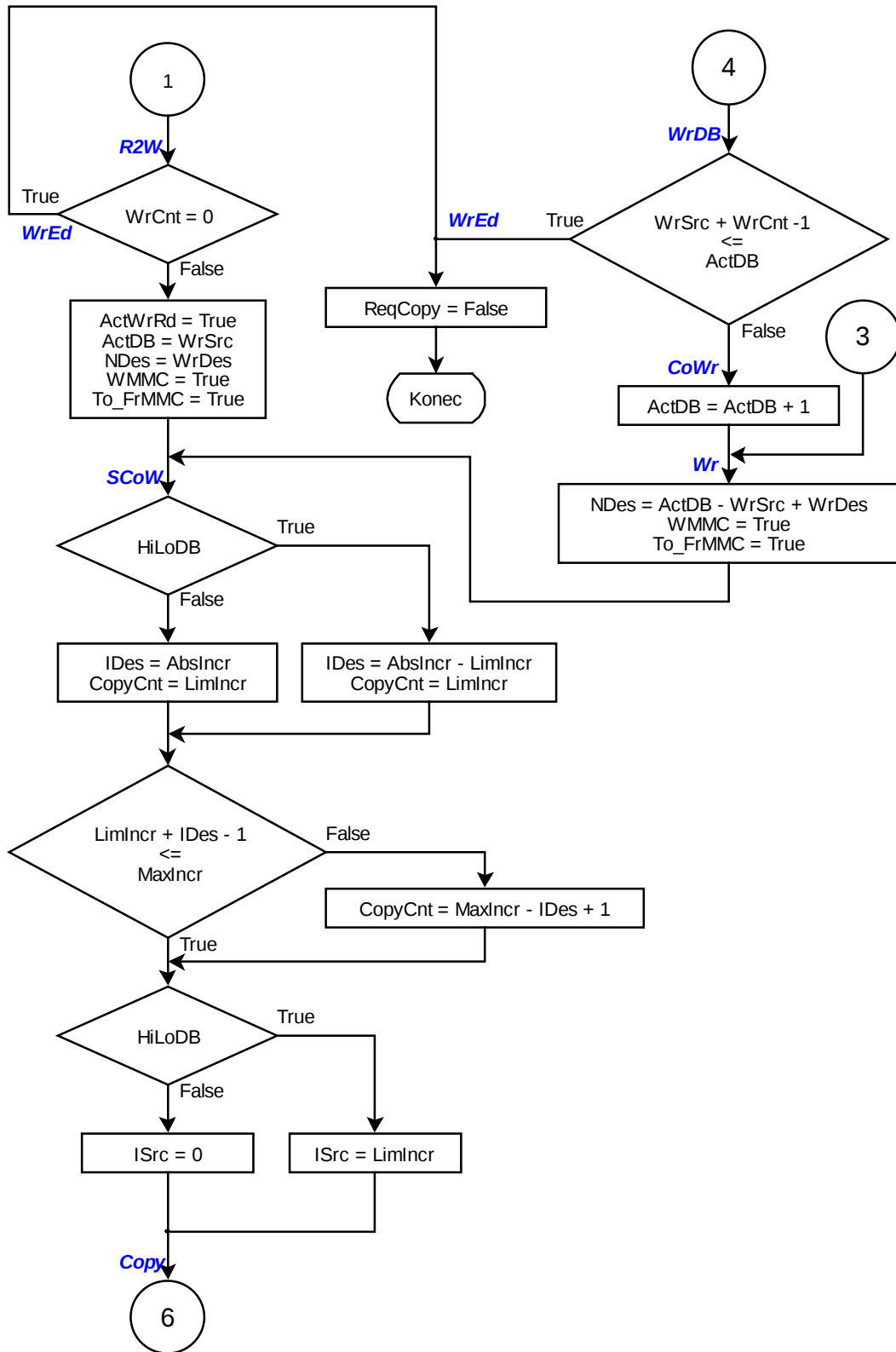
RelIncrMax – vstupně-výstupní Word k nastavení maximální hodnoty ukazatele relativní pozice na hodnotu počtu wordů zbývajících do konce DB bloku v MMC, která je menší než počet wordů v DB bloku ve vnitřní paměti.

Algoritmus funkce:



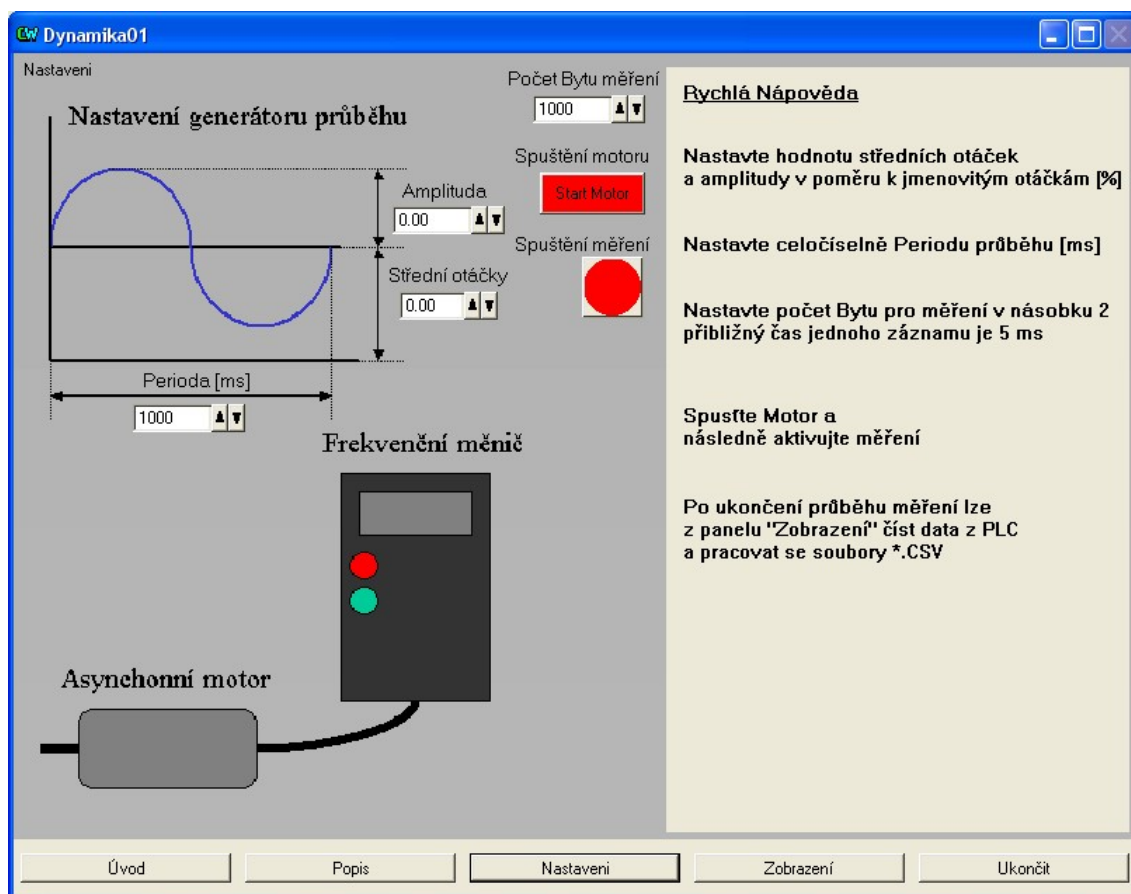






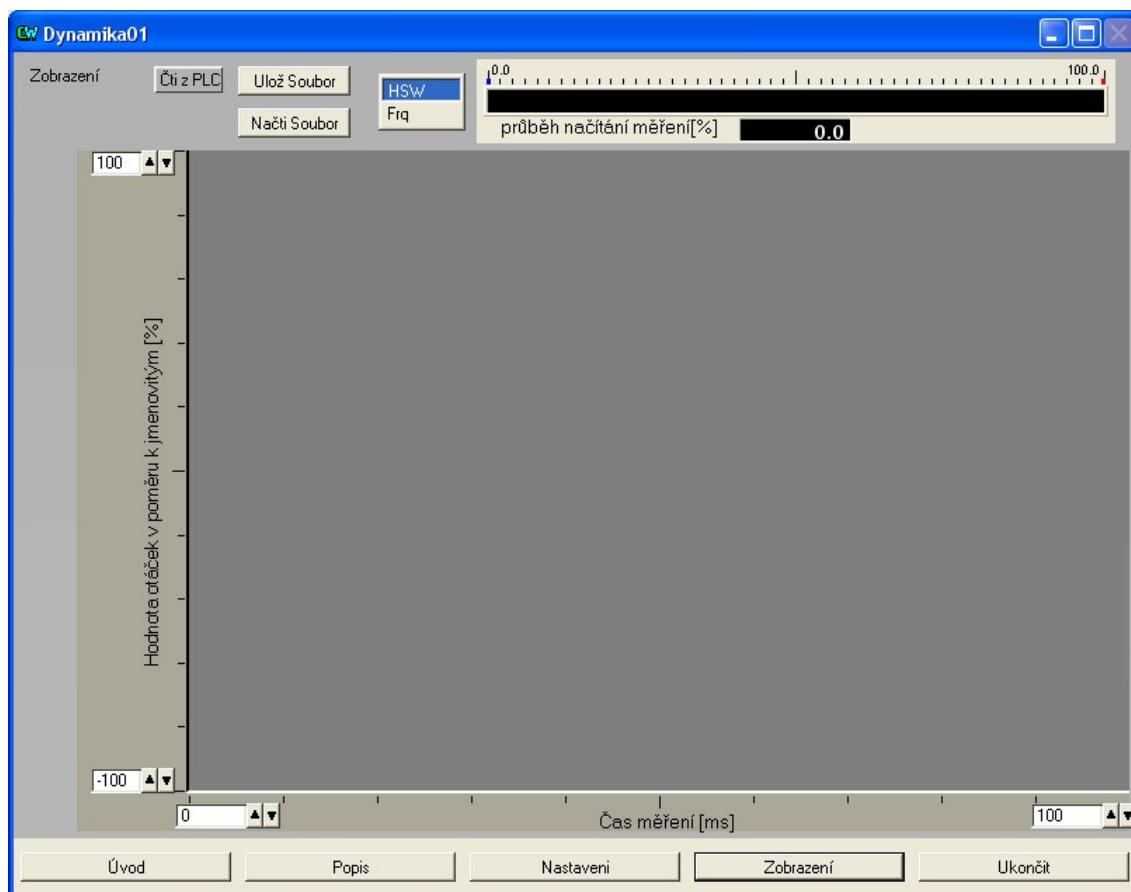
Příloha B –Popis řídicí aplikace

Program má 2 oddělené panely s odlišnou funkcí řídicích prvků. První panel je určen pro nastavení a spuštění měření dynamiky (Obr.B-1).



Obr. B-1: Panel nastavení parametrů průběhu

Tento panel obsahuje ovládací prvky pro nastavení parametrů průběhu. Amplituda a Střední otáčky jsou zadávány jako procentuální část jmenovitých otáček motoru, tyto prvky po změně své hodnoty automaticky přepočítají zadanou hodnotu na rozsah s kterým pracuje PLC automat pro generování průběhu. Hodnota Periody je zadávána v ms a nedochází u ní k přepočtu. „Počet Bytů měření“ udává počet bytů pro měření a vzhledem k 16bitové architektuře PLC je třeba tuto hodnotu zadávat v násobku čísla dvě. Z této hodnoty jsou po spuštění motoru vypočítány hodnoty pro nastavení funkcí realizujících měření, aby při rozběhu měření byly tyto hodnoty již uloženy v proměnných v PLC. Po startu měření je zobrazen informační panel s ukazatelem průběhu měření. PLC automaticky vypne běh měření po zapsání poslední měřené hodnoty, s čímž se zároveň skryje informační panel. Po skončení měření je možné v druhém panelu (Obr. B-2) přečíst z PLC hodnoty měření.



Po stisknutí tlačítka „Čti z PLC“ se spustí postupné načítání hodnot měření, jehož průběh je zobrazován na ukazateli v pravém horním rohu. Po načtení je celé měření zobrazeno v grafu s omezením na zobrazení generovaného nebo měřeného průběhu. Pro zvětšení části grafu je možné zvolit dané limity rozsahu zobrazení. Přepínacími tlačítky „Hsw“ a „Frq“ se přepíná zobrazení mezi žádaným a měřeným průběhem.

Naměřené hodnoty lze uložit do souboru ve formátu CSV přičemž zvětšení zobrazení v grafu nemá vliv na ukládaná data - vždy je uloženo celé měření. Uložené soubory jsou uloženy v aktuálním adresáři aplikace. Případně lze načíst data ze souboru, tím však dojde k přepsání aktuálních hodnot.