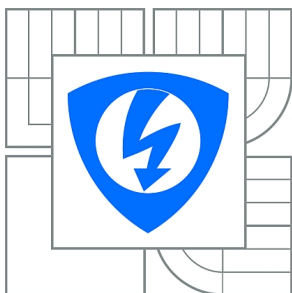


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

IDENTIFIKACE PARAMETRŮ STEJNOSMĚRNÉHO MOTORU

PARAMETER IDENTIFICATION OF DC MOTORS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

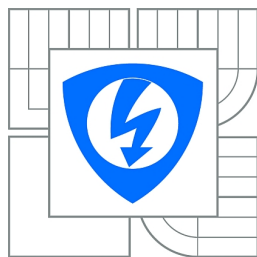
ONDŘEJ KABELKA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. LUDĚK BUCHTA

BRNO 2015



**VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ**

**Fakulta elektrotechniky
a komunikačních technologií**

Ústav automatizace a měřicí techniky

Bakalářská práce

bakalářský studijní obor
Automatizační a měřicí technika

Student: Ondřej Kabelka

ID: 146854

Ročník: 3

Akademický rok: 2014/2015

NÁZEV TÉMATU:

Identifikace parametrů stejnosměrného motoru

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je provést identifikaci parametrů stejnosměrného motoru a návrh řídicího algoritmu.

1. Seznamte se s existujícím modelem dvou spřažených DC motorů, dále se systémem CompactRIO použitým k jeho řízení a prostředím LabView od National Instruments.
2. Navrhněte řídicí algoritmus v prostředí LabView.
3. Proveďte identifikaci parametrů DC motoru (použijte více metod, vzájemné srovnání).
4. Navrhněte laboratorní úlohu zaměřenou na identifikaci parametrů DC motoru a následný návrh regulátorů. Příloha bakalářská práce bude obsahovat vzorové řešení úlohy (Zadání, teoretický rozbor, vlastní vypracování, závěr).

DOPORUČENÁ LITERATURA:

- [1] SODERSTROM, T. STOCIA, P.: System identification. Prentice Hall International Series in Systems and Control Engineering. UK. July 1989. 528s. ISBN 978-0138812362
- [2] NEBORÁK, J.: Modelování a simulace elektrických regulovaných pohonů. VŠB - Technická univerzita Ostrava, 2002.

Termín zadání: 9.2.2015

Termín odevzdání: 25.5.2015

Vedoucí práce: Ing. Luděk Buchta

Konzultanti bakalářské práce:

doc. Ing. Václav Jirsík, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

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

Abstrakt

Tato bakalářská práce řeší problematiku identifikace stejnosměrného motoru. Dále řeší ovládání tohoto motoru a návrh regulátoru pro proudovou smyčku a jeho realizaci. Její součástí je také laboratorní úloha zabývající se tímto tématem.

Klíčová slova

stejnosměrný motor, regulátor, Labview, CompactRIO, proudová smyčka, identifikace, Metoda nejmenších čtverců

Abstract

This bachelor's thesis address the issues of DC motor identification. It also adress the controlling of this motor and creation of current loop regulator and its implementation. Part of the thesis is also laboratory task, which deals with this topic.

Keywords

DC motor, regulator, Labview, CompactRIO, current loop, identification, Least squares method

Bibliografická citace:

KABELKA, O. *Identifikace parametrů stejnosměrného motoru*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2015. 43s. Vedoucí bakalářské práce Ing. Luděk Buchta.

Prohlášení

„Prohlašuji, že svou bakalářskou práci na téma Identifikace parametrů stejnosměrného motoru jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

V Brně dne: **25. května 2015**

.....
podpis autora

Poděkování

Děkuji vedoucímu semestrální práce Ing. Lud'ku Buchtovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne: **25. května 2015**

.....
podpis autora

1.1 Obsah

1.1	Obsah.....	7
1.2	Seznam obrázků	8
1.3	Seznam zkratek	9
2	Úvod.....	11
3	Teoretická část	12
3.1	Program Labview	12
3.2	CompactRIO.....	12
3.3	FPGA.....	13
3.4	DMA kanál	13
3.5	Stejnoseměrný motor	13
3.6	Přechodné jevy ve stejnosměrném motoru.....	15
3.7	Pulzní šířková modulace	17
3.8	Regulátor	17
3.9	Enkodér	18
3.10	Metoda nejmenších čtverců.....	18
3.11	PRBS	20
4	Praktická část	21
4.1	Identifikace motoru za pomoci přechodové charakteristiky	21
4.2	Identifikace motoru pomocí metody nejmenších čtverců	23
4.3	Návrh regulátoru	25
4.4	Program v RT	30
4.5	Program v FPGA.....	33
4.6	Funkce v Matlabu.....	40
5	Závěr	42

1.2 Seznam obrázků

Obr. 3.1: Motor připojený ke CompactRIO, zdroj: vytvořeno autorem	12
Obr. 3.2: Náhradní schéma stejnosměrného motoru, zdroj: vytvořeno autorem	15
Obr. 3.3: Blokové schéma stejnosměrného motoru s konstantním buzením, zdroj: [7].	16
Obr. 3.4: Princip pulzní šířkové modulace, zdroj: [2]	17
Obr. 3.5: Blokové schéma zapojení regulátoru a řízené soustavy, zdroj: [3]	18
Obr. 3.6: Blokové schéma metody nejmenších čtverců, zdroj: 43[8]	19
Obr. 4.1: Průběh napětí převedeného na motor při identifikaci pomocí přechodové charakteristiky, zdroj: vytvořeno autorem	22
Obr. 4.2: Průběh proudu motorem při identifikaci pomocí přechodové charakteristiky, zdroj: vytvořeno autorem	23
Obr. 4.3: Průběh vstupního napětí pro identifikaci metodou nejmenších čtverců, zdroj: vytvořeno autorem	24
Obr. 4.4: Průběh výstupního proudu při identifikaci metodou nejmenších čtverců, zdroj: vytvořeno autorem	25
Obr. 4.5: Identifikace metodou nejmenších čtverců, zdroj: vytvořeno autorem	25
Obr. 4.6: Ukázka nastavování parametrů regulátoru v sisotoolu, zdroj: vytvořeno autorem	26
Obr. 4.7: Odezva na jednotkový skok soustavy s regulátorem (plná čára) a bez regulátoru (přerušovaná čára), zdroj: vytvořeno autorem	27
Obr. 4.8: Průběh napětí v regulované soustavě při zabrzděném motoru, zdroj: vytvořeno autorem	29
Obr. 4.9: Průběh proudu v regulované soustavě při zabrzděném motoru, zdroj: vytvořeno autorem	29
Obr. 4.10: Uživatelské rozhraní RT, zdroj: vytvořeno autorem	31
Obr. 4.11: Výřez programu v RT, 1. část, zdroj: vytvořeno autorem	32
Obr. 4.12: Výřez programu v RT, 2. část, zdroj: vytvořeno autorem	33
Obr. 4.13: Uživatelské rozhraní FPGA, zdroj: vytvořeno autorem	34
Obr. 4.14: Smyčka while v FPGA pro zapnutí a vypnutí motoru, zdroj: vytvořeno autorem	35
Obr. 4.15: Smyčka "PWM Generation", zdroj: vytvořeno autorem	36
Obr. 4.16: Smyčka pro regulaci proudu, 1. část, zdroj: vytvořeno autorem	38
Obr. 4.17: Smyčka pro regulaci proudu, 2. část, zdroj: vytvořeno autorem	38
Obr. 4.18: Smyčka pro regulaci proudu, 3. část, zdroj: vytvořeno autorem	39
Obr. 4.19: Smyčka "Velocity Loop"	39
Obr. 4.20: Smyčka "Encoder Loop", zdroj: vytvořeno autorem	40
Obr. 4.21: Smyčka "PRBS", zdroj: vytvořeno autorem	40

1.3 Seznam zkratek

Značka	Název	Jednotka
FPGA	Field programmable gate array	-
NI cRIO-9076	National Instruments compactRIO	-
USB	Universal serial bus	-
DMA	Direct memory access	-
RAM	Random access memory	-
FIFO	First in first out	-
RT	Real time	-
U_i	Indukované napětí	V
c	Konstanta stroje	-
ω	Otáčky rotoru	min^{-1}
Φ	Magnetický tok	Wb
R_a	Odpor vinutí kotvy	Ω
I_a	Proud kotvou	A
U	Celkové napětí na rotoru	V
L_a	Indukčnost kotvy	H
di_a	Derivace proudu	A
dt	Diferenciál času	s
τ_a	Časová konstanta	s
$F_1(p)$	Přenos systému	-
$F_2(p)$	Přenos systému	-
$F_3(p)$	Přenos systému	-
$F_4(p)$	Přenos systému	-
M	Moment	Nm
M_0	Třecí moment	Nm
J	Moment setrvačnosti	$\text{kg} \cdot \text{m}^2$
p	Laplaceův operátor	-
PWM	Pulzní šířková modulace	-
IGBT	Insulated gate bipolar transistor	-
P	Proporcionální člen	-
I	Integrační člen	-
D	Derivační člen	-
ε	Realizační konstanta	-
$\omega(t)$	Vstupní veličina	-
$e(t)$	Regulační odchylka	-
$x(t)$	Akční veličina	-
$v_1(t)$	Porucha na vstupu soustavy	-
$v_2(t)$	Porucha na výstupu soustavy	-
$y(t)$	Výstup soustavy	-
PRBS	Pseudo random binary sequence	-
T	Časová konstanta	s
F(p)	Přenos motoru	-
MNC	Funkce v Matlabu	-

MNC3	Funkce v Matlabu	-
$F_R(p)$	Výsledný přenos spojitého regulátoru	-
k_R	Konstanta zesílení	-
T_i	Integrační konstanta	-
r_0	Konstanta	-
r_i	Konstanta	-

2 ÚVOD

Cílem této práce je vytvořit funkční program v prostředí LabView, ve kterém by se daly použít různé metody identifikace stejnosměrného motoru, na který budou následně navrženy vhodné regulátory.

Z velkého množství používaných motorů jsou stejnosměrné motory nejvhodnější k identifikaci a následné regulaci, díky jejich lineární závislosti otáček na napětí.

LabView je grafický programovací jazyk vyvinutý společností National Instruments. Může se používat při vývoji testovacích, měřicích a řídicích aplikací. Díky němu lze provádět celou řadu simulací a analýz, navíc je v tomto programu také možnost ověřit funkčnost regulátorů.

V závěru práce bude navrhnutá a vypracována laboratorní úloha na téma identifikace stejnosměrného motoru, ve které si studenti budou moci zkusit identifikaci a návrh regulátoru ke skutečnému motoru. Laboratorní úloha obsahuje zadání, postup měření a návod ke zpracování naměřených výsledků.

3 TEORETICKÁ ČÁST

3.1 Program Labview

Program Labview je grafické vývojové prostředí od firmy National Instruments. Programování probíhá za pomoci bloků, které reprezentují určité operace, a které se v blokovém diagramu zapojují do sebe. Čelní panel představuje rozhraní pro uživatele programu, který může ovládat za pomoci grafických prvků a řídit tak chod celého programu.

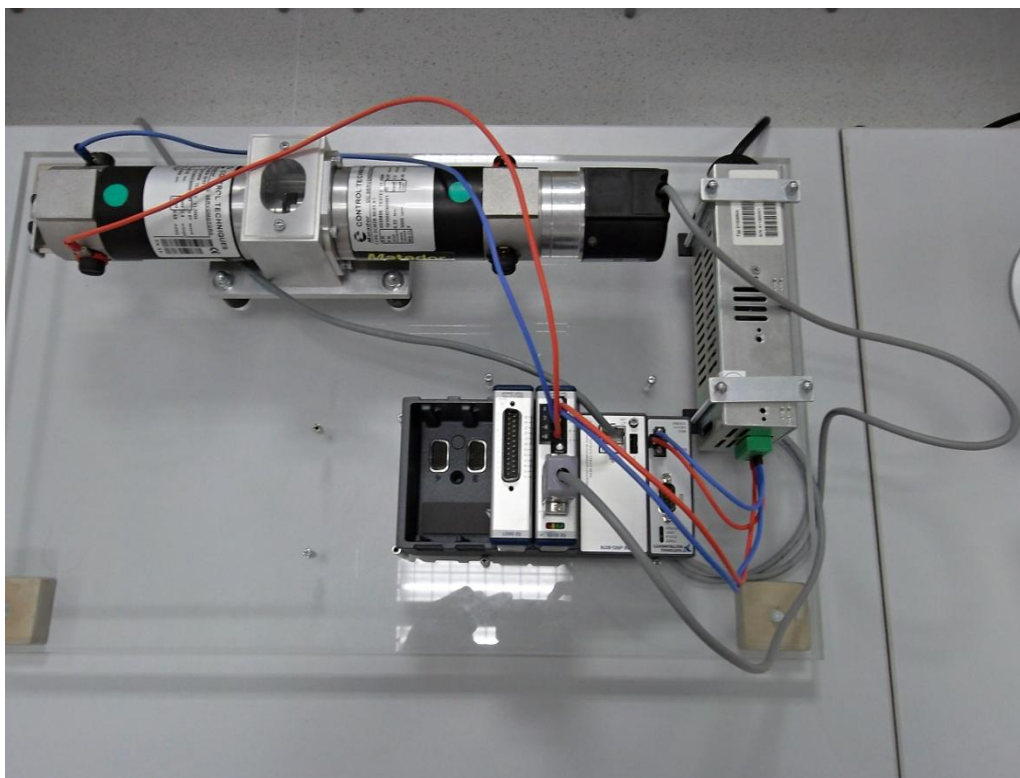
Výhodou Labview je jednoduché ovládání, které napomáhá rychlému pochopení principů programování s ním. Nevýhodou je fakt, že u rozsáhlejšího programu se blokový diagram může snadno stát poměrně nepřehledným a tím i špatně čitelným.

3.2 CompactRIO

CompactRIO je rekonfigurovatelná architektura od společnosti National Instruments pro sběr dat. Robustní architektura zahrnuje I/O moduly, rekonfigurovatelné FPGA šasi a kontroler. CompactRIO se programuje za pomoci Labview po sběrnici Ethernet 10/100 Mb/s. [1] V práci byly použity následující moduly:

NI cRIO-9076: Programovatelný kontrolér obsahující FPGA pole s frekvencí 400 MHz. Rozsah pracovní teploty -20 až 50 °C. Má 4 sloty pro přídavné moduly, 10/100 BASE-T Ethernet port, USB 2.0 a RS232 serial port pro připojení k perifériím.

NI9505: Karta určená pro řízení motoru napájecím napětím. Výstupní proud je max. 5A při 40 °C nebo 1A při 70 °C a 30V. Má zabudované čidlo proudu a rozhraní pro enkodér.



Obr. 3.1: Motor připojený ke CompactRIO, zdroj: vytvořeno autorem

3.3 FPGA

Zkratka FPGA znamená “Field Programmable Gate Array“, tedy programovatelné hradlové pole. FPGA při programování fyzicky přepojuje číslicové obvody, což velmi zrychluje následný běh programu. Pole se dá kdykoliv znovu přeprogramovat a tím se do něj dají nahrát nové programy. Velikost programu je omezena počtem programovatelných buněk pole. Na trhu jsou modely s různými velikostmi polí. Nevýhodou dlouhého nahrávání programu (v řádech desítek minut pro větší programy) vyvažuje rychlost vytvořeného programu, univerzálnost a možnost kdykoliv pole přeprogramovat.

3.4 DMA kanál

DMA (Direct Memory Access – přímý přístup k paměti) umožňuje FPGA přístup a využívání RAM paměti procesoru, který funguje jako hostitel. To umožňuje vyšší rychlost přenosu dat z FPGA do hostitelského počítače.

Data jsou přenášena architekturou FIFO (First In, First Out), která se skládá ze dvou částí. První část FIFO je v FPGA a využívá paměť RAM zařízení FPGA, druhá část je v počítači a využívá jeho paměť. DMA kanál pak přenáší data přímo z RAM paměti FPGA do paměti počítače. [4]

V této práci se DMA kanál používá pro přenos velkého množství naměřených dat z FPGA do počítače. V FPGA se měří hodnota proudu každých 50 μ s, smyčka v RT ale probíhá každých 100 ms. Z tohoto důvodu je třeba při každém průchodu smyčky v RT vyčíst z FPGA 2000 hodnot naměřeného proudu. Z tohoto důvodu má DMA nastavenou velikost bufferu na 4095 hodnot, aby se zamezilo jeho přetečení.

3.5 Stejnosměrný motor

Stejnosměrné motory jsou prakticky nejstarší používané motory. Přestože nejsou konstrukčně nejjednodušší (nejjednodušší na konstrukci je asynchronní motor s kotvou nakrátko), mají obrovskou výhodu oproti ostatním motorům vyplývající z dobré regulace. Takový stejnosměrný motor tvoří ideální regulační systém, protože pouhou změnou velikosti napětí přiváděného na kotvu můžeme plynule řídit otáčky motoru, změnou polaritu můžeme měnit směr otáček motoru. Tyto motory nacházejí svoje využití především tam, kde je potřeba velký rozsah regulace otáček nebo v místech s těžkými podmínkami rozběhu. Používají se například v obráběcích strojích, průmyslových robotech a nacházejí svoje využití i v automobilovém průmyslu.

Existuje řada možností, jak zapojit kotvu (rotor) stejnosměrného motoru, abychom vybudili magnetický tok kolem cívek kotvy. K vytvoření potřebného magnetického pole můžeme použít například permanentní magnety, cizí buzení, sériové buzení, paralelní buzení anebo kompaundní (séριοvě-paralelní) buzení. Pro potřeby regulace soustavy se stejnosměrným motorem se používá nejčastěji motor s cizím buzením nebo s permanentními magnety na statoru. K výrobě magnetů se používají magneticky tvrdé materiály, ferity a materiály ze vzácných zemin. Těmi mohou být například sloučeniny neodymu, železa, niklu, kobaltu a bóru.

Konstrukčně motor obsahuje stator a rotor. Na statoru jsou permanentní magnety a pólové nástavce z magneticky měkkého železa. Pólové nástavce slouží ke koncentraci magnetického toku do vzduchové mezery. Rotor obsahuje vinutí cívky nebo více cívek

uložené do drážek rotoru, konce cívek jsou přivedeny na komutátor. Ten společně s uhlíkovými kartáči tvoří sběrné ústrojí.

Stejnoseměrné motory bývají popsány základními rovnicemi, které vyjadřují děje uvnitř motoru. Mezi ně patří například rovnice indukovaného napětí na kotvě a momentová rovnice. Rovnice vyjadřující indukované napětí na vinutí kotvy, velikost napětí závisí především na rychlosti rotoru ω

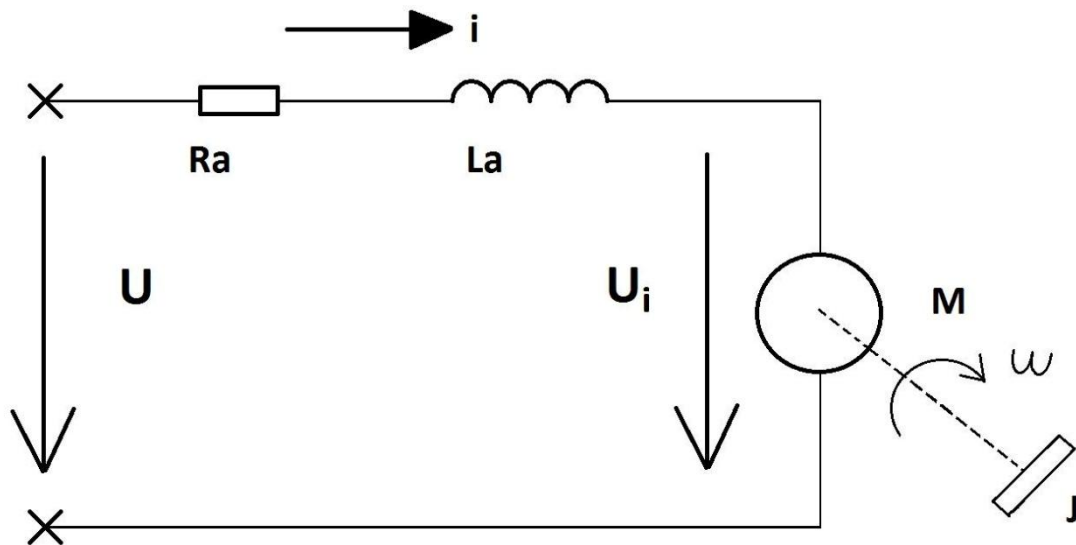
$$U_i = c \cdot \phi \cdot \omega \quad (3.1)$$

Proud procházející kotvou v reakci na magnetický tok permanentních magnetů vyvolá moment na hřídeli

$$M = c \cdot \phi \cdot I \quad (3.2)$$

Stejnoseměrný motor s permanentními magnety má konstantní magnetický tok závislý na použitých magnetech a na konstrukci magnetického obvodu. Ve dvou předešlých rovnicích výraz $c \cdot \phi$ vystupuje jako konstanta závislá právě na konstrukci motoru. Kvůli tomu lze otáčky motoru řídit pouze změnou velikosti napětí na kotvě. Pro napětí na kotvě v ustáleném stavu platí rovnice sčítající indukované napětí a úbytky napětí na odporu R_a

$$U = R_a \cdot I_a + c \cdot \phi \cdot \omega = R_a \cdot I_a + U_i \quad (3.3)$$



Obr. 3.2: Náhradní schéma stejnosměrného motoru, zdroj: vytvořeno autorem

3.6 Přechodné jevy ve stejnosměrném motoru

Rovnice 2.3 platí pouze pro ustálený stav motoru. Náhradní schéma zobrazené na obrázku Obr. 3.2 slouží k detailnějšímu popisu dynamických dějů v motoru. Díky tomuto schématu můžeme vytvořit soustavu diferenciálních rovnic a ty pak díky Laplaceově transformaci převést na operátorový přenos systému.

$$u_a = R_a \cdot I_a + L_a \cdot \frac{di_a}{dt} + c \cdot \phi \cdot \omega \quad (3.4)$$

$$M = c \cdot \phi \cdot i_a = J \cdot \frac{d\omega}{dt} + B \cdot \omega + M_0 \quad (3.5)$$

V těchto rovnicích je L_a indukčnost vinutí kotvy, J je moment setrvačnosti rotoru a hřídele. B je koeficient tření a M_0 je moment zátěže. Rovnice 2.4 je napěťová rovnice kotvy, druhá rovnice 2.5 se nazývá momentová rovnice a vyjadřuje rovnost momentů na hřídeli. Z obou rovnic je možné vyjádřit přenosovou funkci díky Laplaceově transformaci převedením derivace proudu na operátor p . Celá rovnice v operátorovém tvaru pak bude

$$U_a = R_a \cdot I_a + p \cdot L_a \cdot I_a + U_i \quad (3.6)$$

Odpor a indukčnost kotvy se dá vyjádřit operátorovým přenosem $F_1(p)$. Ten reprezentuje poměr výstupního proudu kotvou ku rozdílu napětí přivedeného na kotvu a indukovaného napětí na kotvě. Přenos těchto obvodových prvků je setrvačný člen se zesílením $\frac{1}{R_a}$, časová konstanta je τ_a .

$$F_1(p) = \frac{I_a}{U_a - U_i} = \frac{\frac{1}{R_a}}{1 + p \cdot \tau_a} \quad (3.7)$$

$$\tau_a = \frac{L_a}{R_a} \quad (3.8)$$

Na schématu zobrazený model obsahuje ještě proporcionální člen vyjadřující závislost momentu stejnosměrného motoru na proudu procházejícího kotvou motoru. Dynamický moment na hřídeli je $m = c \cdot \phi \cdot i_a$, po jeho převedení na operátorový tvar je to $M = c \cdot \phi \cdot I_a$. Přenos takového členu je dán vztahem

$$F_2(p) = \frac{M}{I_a} = c \cdot \phi \quad (3.9)$$

Při zanedbání členu reprezentujícího tření můžeme psát rovnici momentové rovnováhy v operátorovém tvaru

$$M = p \cdot J \cdot \omega + M_0 \quad (3.10)$$

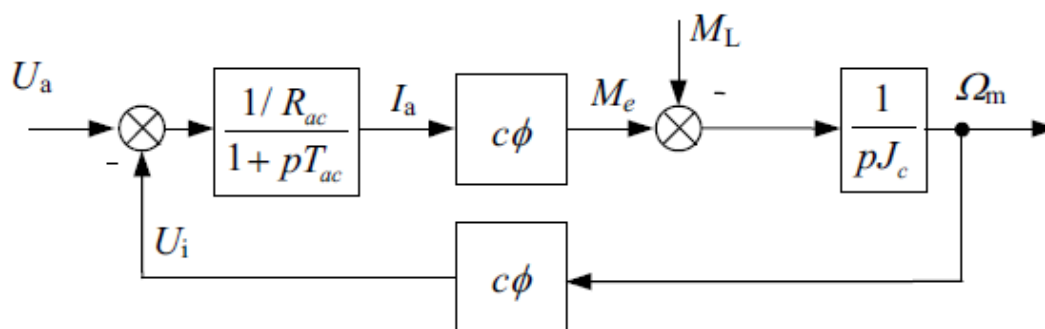
Člen reprezentující moment setrvačnosti má integrační charakter a jeho přenos je

$$F_3(p) = \frac{\omega}{M - M_0} \quad (3.11)$$

Napětí indukované v kotvě je společně s úhlovou rychlostí vyjádřeno jako proporcionální přenosový člen a má tvar

$$F_4(p) = \frac{U_i}{\omega} = c \cdot \phi \quad (3.12)$$

Pomocí všech výše uvedených přenosových funkcí můžeme vytvořit matematický model stejnosměrného motoru s konstantním magnetickým tokem. Schéma matematického modelu je na obrázku Obr. 3.3.



Obr. 3.3: Blokové schéma stejnosměrného motoru s konstantním buzením, zdroj: [7]

Stejnoseměrný motor je ideální regulační soustava. Jeho výstup, tedy otáčky lineárně závisí na vstupu, což je napětí.

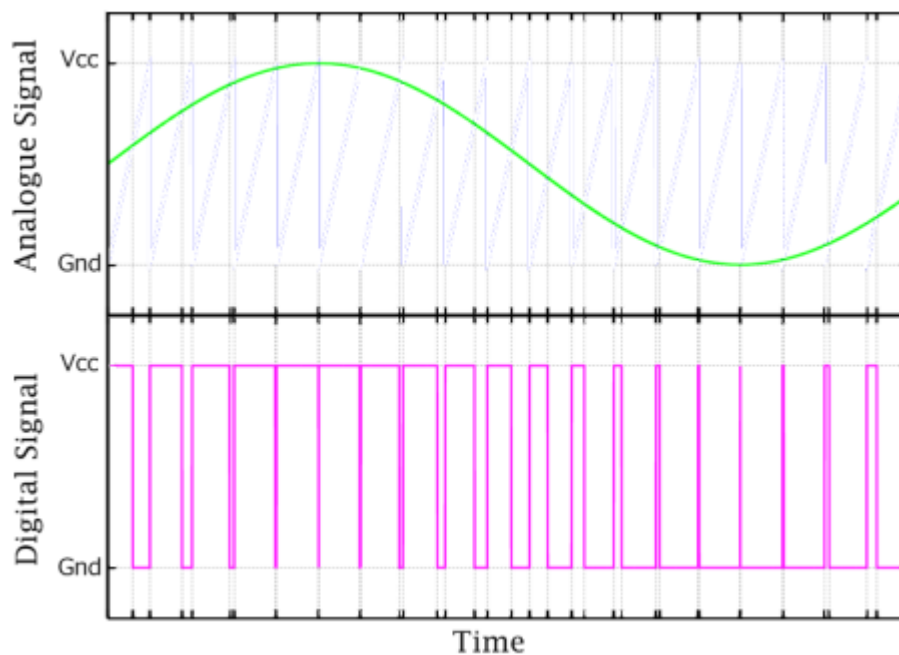
3.7 Pulzní šířková modulace

Pulzní šířková modulace, (anglicky Pulse-width modulation, zkráceně PWM) je modulace signálu, která umožňuje přenášení hodnoty signálu pouze za pomoci logické 1 a 0. Modulovací signál má periodický pilovitý průběh. Modulovaný signál se porovnává se signálem modulovacím a na výstup se posílá logická jednička, pokud má modulovaný signál vyšší hodnotu než modulovací, a logická nula, pokud je modulovaný signál menší než modulovací. Signál na výstupu má proto pouze hodnoty logické 1 a 0.

V rámci jedné periody modulovacího signálu se pak určuje tzv. střída – poměr mezi dobou, kdy byla na výstupu logická jednička k době, kdy byla na výstupu logická nula. Střída nabývá hodnot 0 – 100%, kdy 25% udává, že hodnota výstupního napětí bude rovna čtvrtině vstupního napětí, za předpokladu, že maximální hodnota vstupního signálu odpovídá maximální hodnotě signálu výstupního.

Ve své práci využívám PWM k přenosu velikosti napájecího napětí motoru. PWM má dále využití u kódování informací ve sdělovací technice.

Nevýhodou PWM jsou ztráty způsobené spínáním a rozepínáním tranzistoru. Ztráty závisí na době sepnutí – čím delší jsou doby spínání a rozepínání, tím větší jsou i ztráty. Z tohoto důvodu se velmi často využívá tranzistor IGBT, který má nižší ztrátový výkon, než jiné tranzistory. Využívá se do střídačů a měničů.



Obr. 3.4: Princip pulzní šířkové modulace, zdroj: [2]

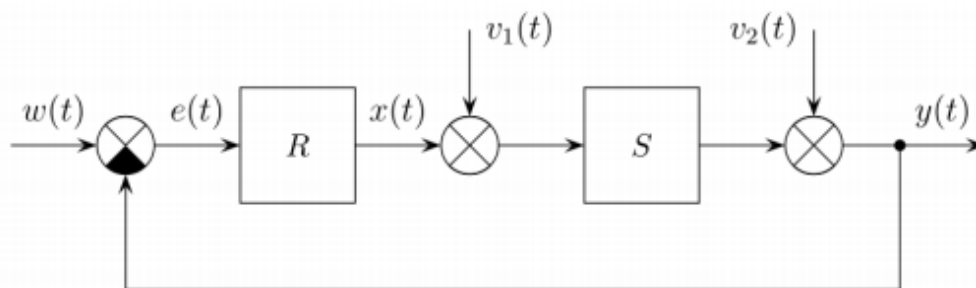
3.8 Regulátor

Regulátor je zařízení, které působí na regulovanou soustavu pomocí akční veličiny tak, aby odchylka soustavy od požadované hodnoty byla co nejmenší. Regulátory mají tři základní složky: proporcionální, integrační a derivační, tedy P, I a D.

Proporcionální složka je zesílení regulátoru. U P regulátoru je akční veličina přímo úměrná regulační odchylce. Nevýhodou P regulátoru je, že nedokáže zajistit nulovou ustálenou odchylku žádané veličiny při působení poruchy na vstupu soustavy.

Integrační složka zajišťuje schopnost regulátoru vyregulovat poruchu, ale zhoršuje dynamické vlastnosti regulátoru tím, že zpomaluje přechodný děj.

Derivační složka zrychluje regulační děj, avšak D regulátor není sám o sobě použitelný, protože nesplňuje vyregulování na nízkých frekvencích. Z tohoto důvodu se derivační složka používá pouze v PD a PID regulátorech. V praxi se k D složce regulátoru musí přidat realizační konstanta ϵ , protože D složka přidává do regulátoru nulu, což vede k fyzikálně nerealizovatelnému regulátoru s vyšším řádem čitatele než jmenovatele. [3]



Obr. 3.5: Blokové schéma zapojení regulátoru a řízené soustavy, zdroj: [3]

Na obrázku Obr. 3.5 je zobrazeno blokové schéma obecného zapojení soustavy a regulátoru, kde $w(t)$ je žádaná hodnota (vstupní veličina), $e(t)$ je regulační odchylka (rozdíl mezi žádanou a výstupní hodnotou), $x(t)$ je akční veličina, kterou regulátor působí na soustavu, $v_1(t)$ je porucha na vstupu soustavy, $v_2(t)$ je porucha na výstupu soustavy a $y(t)$ je výstup soustavy, tedy regulovaná veličina.

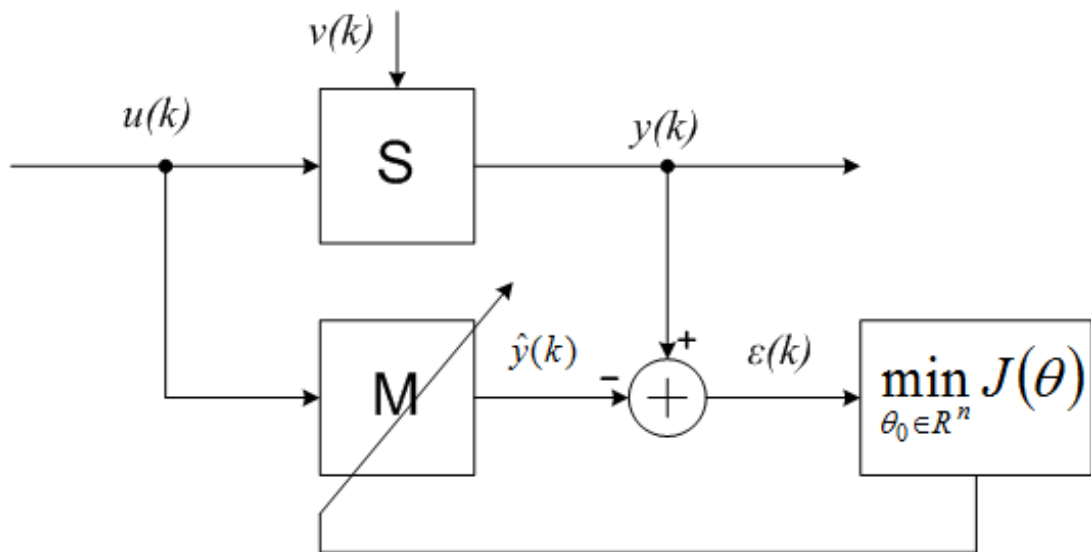
3.9 Enkodér

Enkodér v této práci vyhodnocuje pozici a rychlost rotoru za pomoci optoelektronického senzoru posuvu, který je v kartě NI9505. Princip tohoto senzoru spočívá v clonění světelného toku mezi zdrojem a fotocitlivými prvky pravítkem (kotoučem). Posuv rotoru o 1 krok vyvolá přerušení světelného svazku a výstupní signál fotocitlivého detektoru po úpravě na impuls unifikovaného tvaru inkrementuje obsah čítače. Měřený posuv je tedy dán obsahem čítače vynulovaného při referenční poloze.[5]

3.10 Metoda nejmenších čtverců

Tato metoda vychází z použití matematické metody s názvem lineární regrese. Tato metoda se snaží o jednoduchou aproximaci parametrů modelu za použití předložených zdrojových dat.

Z matematického hlediska je možné říci, že daná metoda minimalizuje velikost odchylky mezi výstupem soustavy $y(t)$ a výstupem modelu $y_m(t)$. To je názorně vidět na blokovém schématu zobrazeném na obrázku níže. [8]



Obr. 3.6: Blokové schéma metody nejmenších čtverců, zdroj:43[8]

Modelem v pojetí identifikace pomocí MNČ budeme rozumět rovnici, která za pomoci znalosti o zpožděných hodnotách vstupů a výstupů, a za pomoci parametrů příslušejících jedné konkrétní realizaci modelu dá výstupní hodnotu na výstupu dané soustavy.

$$\hat{y}(k) = \sum_{i=0}^M b_i u(k-i) - \sum_{j=0}^N a_j y(k-j) \quad (3.13)$$

Pro aplikaci metody nejmenších čtverců je vhodné volit jako model rovnici, zadanou ve tvaru:

$$\hat{y}(k) = [b_0 \quad b_1 \quad \dots \quad b_M] \begin{bmatrix} u(k) \\ u(k-1) \\ \dots \\ u(k-M) \end{bmatrix} - [a_1 \quad a_2 \quad \dots \quad a_N] \begin{bmatrix} y(k-1) \\ y(k-2) \\ \dots \\ y(k-M) \end{bmatrix} \quad (3.14)$$

Tuto rovnici lze formálně přepsat za pomoci maticového násobení jako:

$$\theta = \begin{bmatrix} b_0 \\ \dots \\ b_N \\ a_1 \\ \dots \\ a_N \end{bmatrix} \quad (3.15)$$

$$\varphi(k) = \begin{bmatrix} u(k) \\ \dots \\ u(k-M) \\ -y(k-1) \\ \dots \\ -y(k-N) \end{bmatrix} \quad (3.16)$$

$$\Phi = \begin{bmatrix} \varphi(k)^T \\ \dots \\ \varphi(P)^T \end{bmatrix} \quad (3.17)$$

$$Y = \begin{bmatrix} y(k) \\ \dots \\ y(P) \end{bmatrix} \quad (3.18)$$

$$\hat{y}(k) = \varphi(k)^T \theta \quad (3.19)$$

$$Y = \Phi \theta \quad (3.20)$$

Z tohoto maticového zápisu lze již jednoduše odvodit vlastní výpočet metody
nejmenších čtverců

$$\Phi^T Y = (\Phi^T \Phi) \theta \quad (3.21)$$

3.11 PRBS

Pseudo náhodná binární posloupnost (Pseudo Random Binary Sequence) je posloupnost podobná bílému šumu, lze však vždy určit její hodnotu v následujícím kroku. Posloupnost je binární, může proto nabývat pouze dvou hodnot. Generuje se za pomoci sčítaček modulo 2 a posuvných registrů. V PRBS se postupně vystřídají všechny stavy s výjimkou samých nul. Pro identifikaci systému je nutné, aby délka nejdelšího impulzu byla delší, než doba náběhu systému. [8]

4 PRAKTICKÁ ČÁST

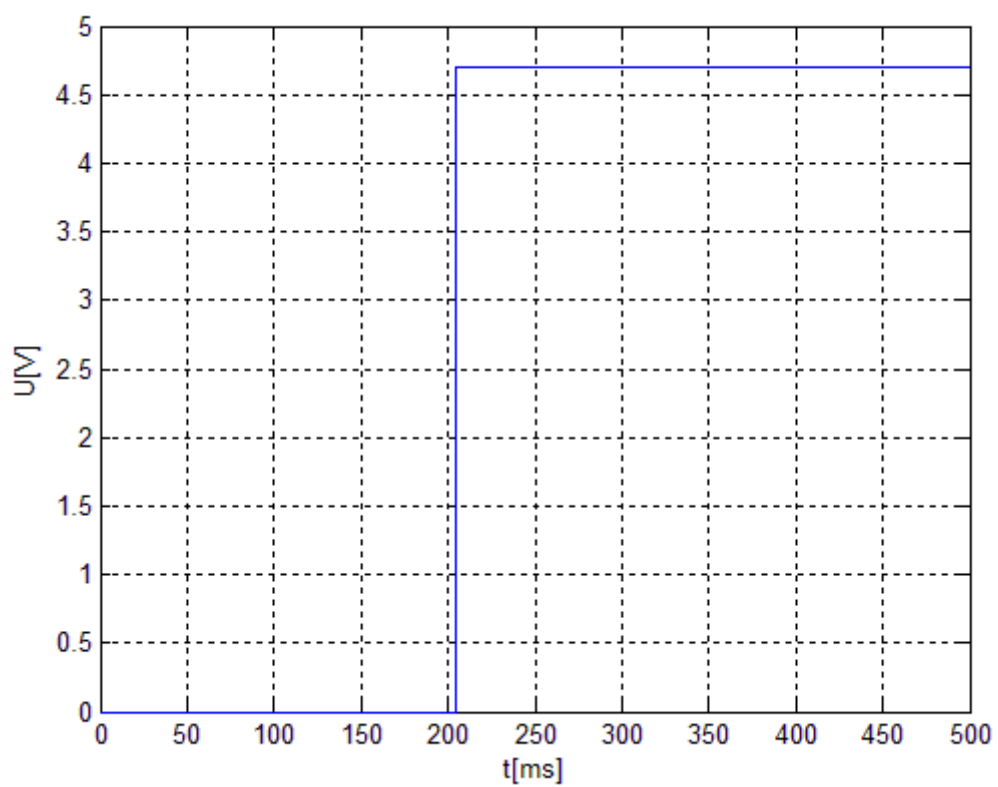
4.1 Identifikace motoru za pomoci přechodové charakteristiky

Samotná identifikace soustavy se spustí stisknutím tlačítka “identifikovat” v RT. Před začátkem identifikace je třeba se ujistit, že v grafech proudu a napětí v RT není zaznamenaný žádný průběh a v případě, že je, je třeba na ně kliknout pravým tlačítkem myši a zvolit možnost “clear chart”. Jestliže budeme chtít identifikovat pouze motor bez regulátoru, musíme ještě v FPGA stisknout tlačítko “vypnout regulátor”. Tím zajistíme, že motor bude ovládán pouze nastavením Duty cycle v RT.

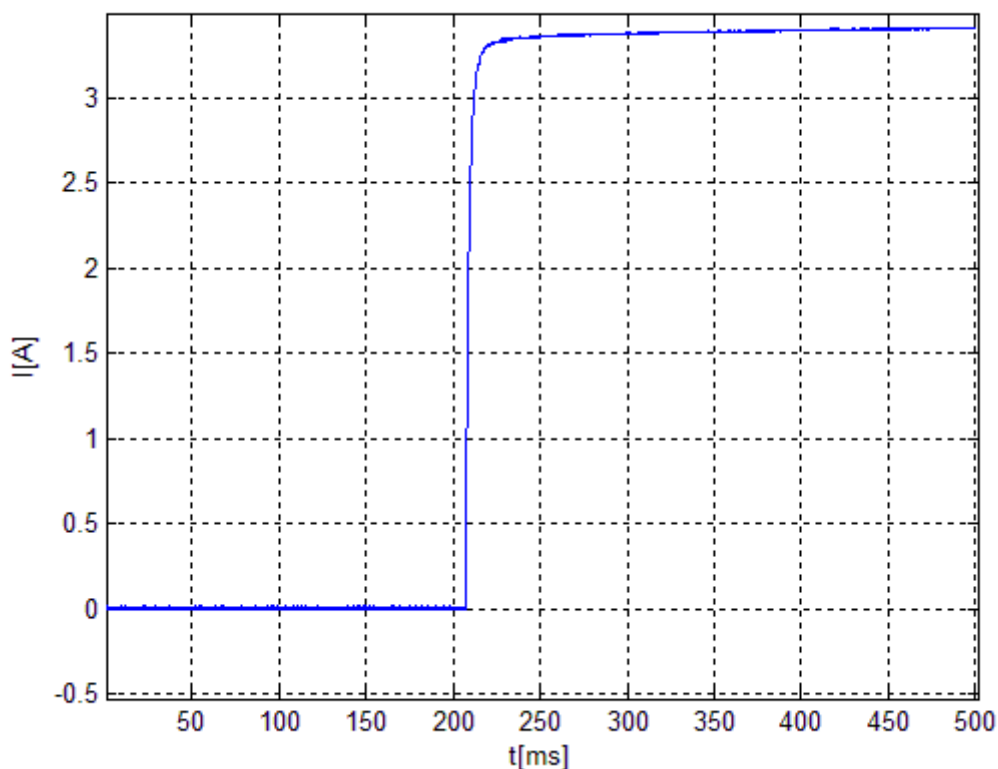
Jakmile je nastavený software, je ještě potřeba zajistit hřídel motoru proti otáčení. K tomuto účelu slouží přípravek vytvořený z malého plíšku, do kterého byly vyvrtány otvory pro šroubky a který se přišroubuje na hřídel motoru. Při identifikaci metodou Locked rotor test je třeba zajistit hřídel takovým způsobem, aby se po přivedení napětí necukla, protože ono cuknutí by se negativně projevilo na přechodové charakteristice proudu. Je možné, že se charakteristiku nepovede naměřit napoprvé, a bude třeba měření opakovat. V tom případě je třeba znovu vyčistit grafy napětí a proudu v RT.

Je vhodné provést několik měření přechodové charakteristiky a potom udělat jejich průměr např. metodou nejmenších čtverců.

Tlačítko “identifikovat” spustí v programu 0,5s trvající sekvenci, která nejprve nastaví duty cycle na 0 a poté přivede na vstup motoru jednotkový skok o velikosti 40% maximálního napětí, tedy duty cycle = 800. Celý průběh je zakreslen do grafů proudu a napětí, které přijímají data jen během této sekvence, aby se charakteristiky nepřepsaly navazujícím příchozím signálem.



Obr. 4.1: Průběh napětí převedeného na motor při identifikaci pomocí přechodové charakteristiky, zdroj: vytvořeno autorem



Obr. 4.2: Průběh proudu motorem při identifikaci pomocí přechodové charakteristiky, zdroj: vytvořeno autorem

Charakteristiky z grafů se vyexportují do excelu, kde se uloží jako textové soubory s oddělovači, a poté se nahrají do Matlabu. V Matlabu se na obě charakteristiky zavolá funkce “identifikace“, která má jako vstupní argumenty právě průběhy napětí a proudu. Výstupem funkce jsou zesílení K a časová konstanta T , které lze dosadit do přenosu stejnosměrného motoru

$$F(p) = \frac{K}{1 + Tp} \quad (4.1)$$

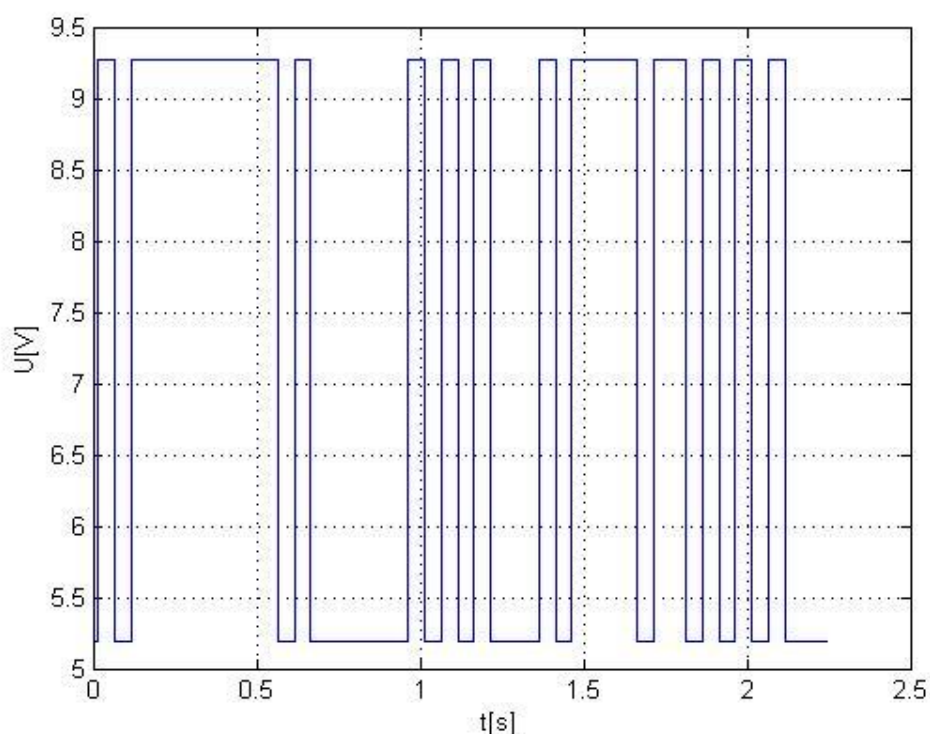
Přenos motoru vyšel v tomto případě $F(p) = \frac{0,7255}{1 + 0,0032p}$

4.2 Identifikace motoru pomocí metody nejmenších čtverců

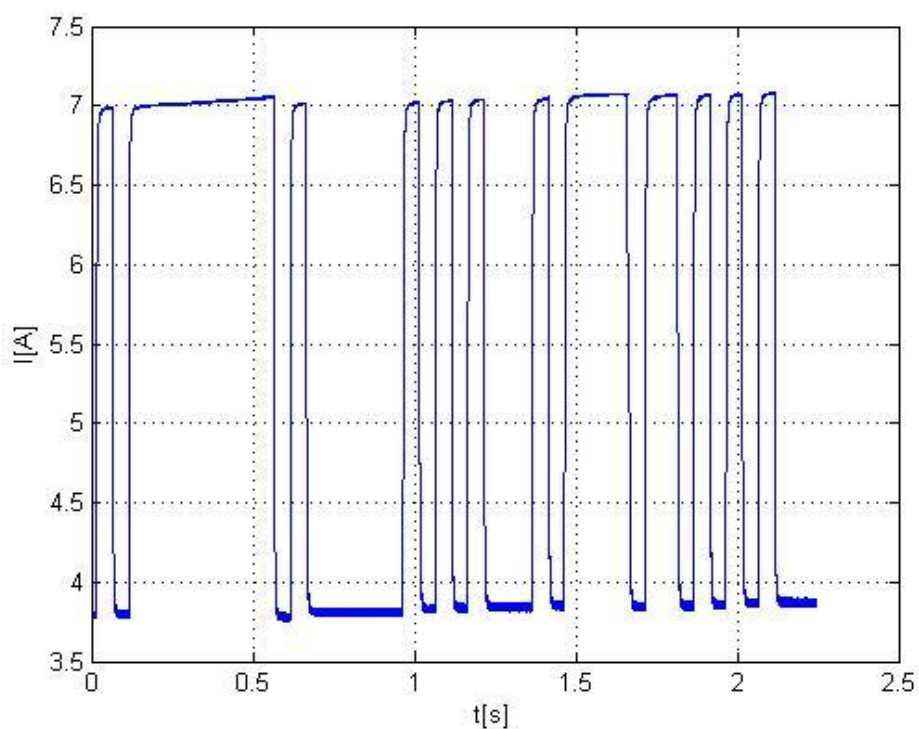
Nejprve je třeba naměřit odezvu systému na PRBS. Ta se naměří tak, že se v FPGA vypnou oba regulátory tlačítky “Disable current regulator“ a “Disable velocity regulator“. Poté je třeba přepnout vstupní signál tlačítkem “PRBS“, nastavit frekvenci PRBS a maximum a minimum tohoto signálu v příslušných kolonkách. Frekvence smyčky se nastavuje v μs , a pro měření byla nastavena na 50 000 μs , tedy 50 ms.

Jakmile jsou nastaveny žádané hodnoty, v RT se spustí motor tlačítkem “Enable drive“. Průběh vstupního napětí i výstupního proudu pak lze vyčíst z příslušných grafů. Průběhy z grafů je třeba exportovat po pauze programu, aby data z napětí a proudu odpovídala stejnému časovému rozmezí.

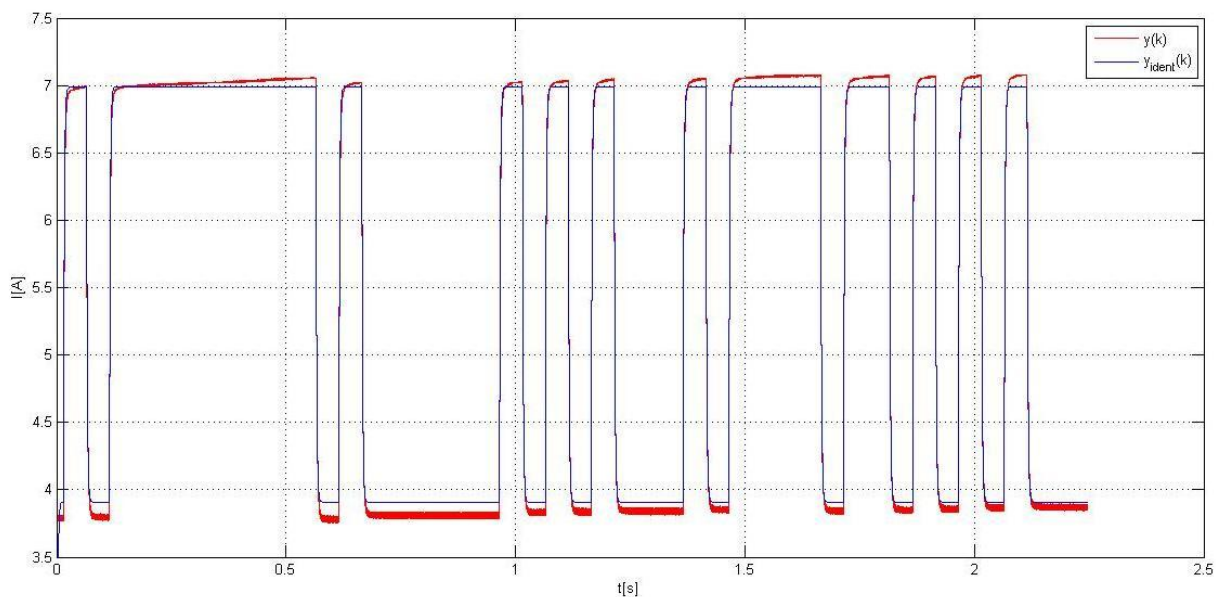
Vyexportovaná data se poté zpracují v programu Matlab. Pro zpracování byly vytvořeny dvě funkce, MNC a MNC3. Obě funkce jsou velmi podobné, jejich výstupem je spojitý přenos, vstupními argumenty jsou vstup a výstup ze systému. Obě funkce zpracovávají data metodou nejmenších čtverců, výsledkem funkce MNC je spojitý přenos systému prvního řádu, výsledkem funkce MNC3 je spojitý přenos systému 3. řádu. Pro identifikaci motoru byla použita funkce MNC a výsledný přenos vyšel $F(p) = \frac{0,7309}{0,0015p+1}$



Obr. 4.3: Průběh vstupního napětí pro identifikaci metodou nejmenších čtverců, zdroj: vytvořeno autorem



Obr. 4.4: Průběh výstupního proudu při identifikaci metodou nejmenších čtverců, zdroj: vytvořeno autorem



Obr. 4.5: Identifikace metodou nejmenších čtverců, zdroj: vytvořeno autorem

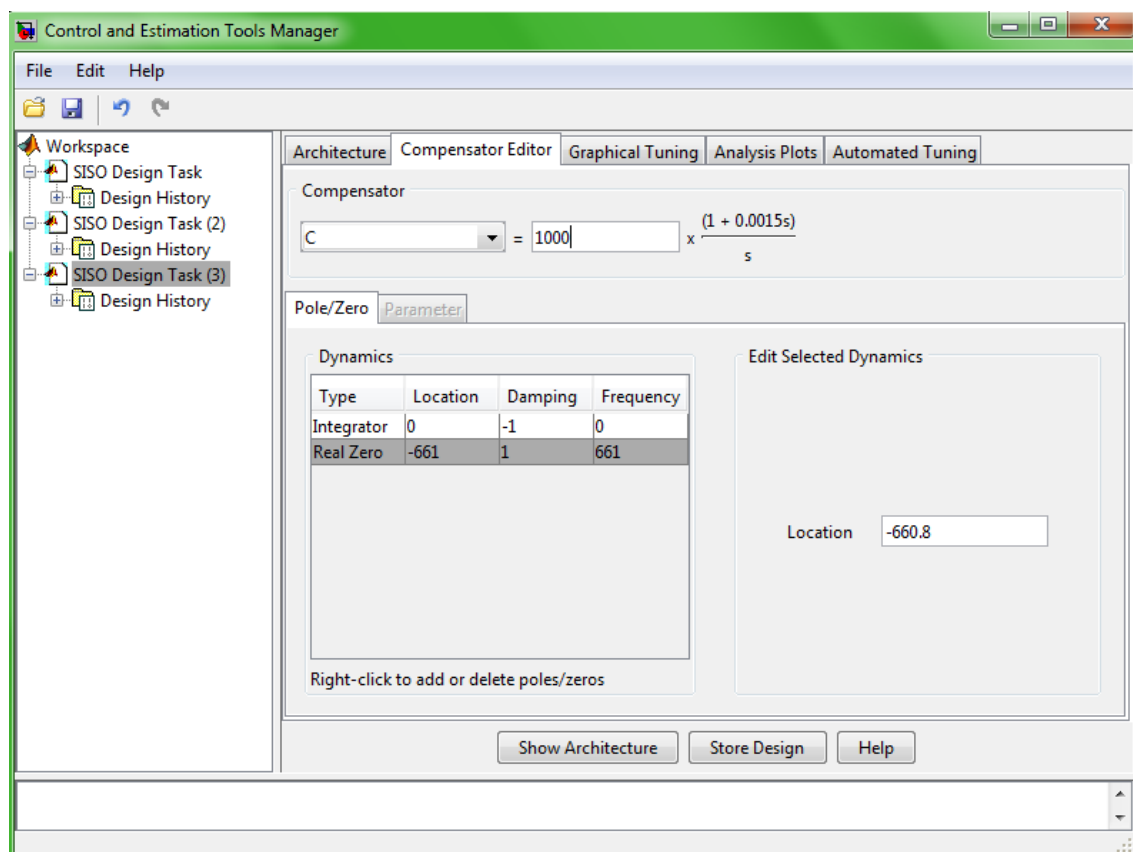
Na obrázku Obr. 4.5 je vidět průběh naměřeného výstupního proudu (červeně) a průběh výstupu vypočítaného modelu (modře). Lze vidět, že model kopíruje naměřená data velmi přesně.

4.3 Návrh regulátoru

Pro návrh regulátoru lze použít nástroj sisotool v Matlabu. Samotný návrh je prováděn metodou frekvenčních charakteristik. Identifikaci pomocí přechodové charakteristiky

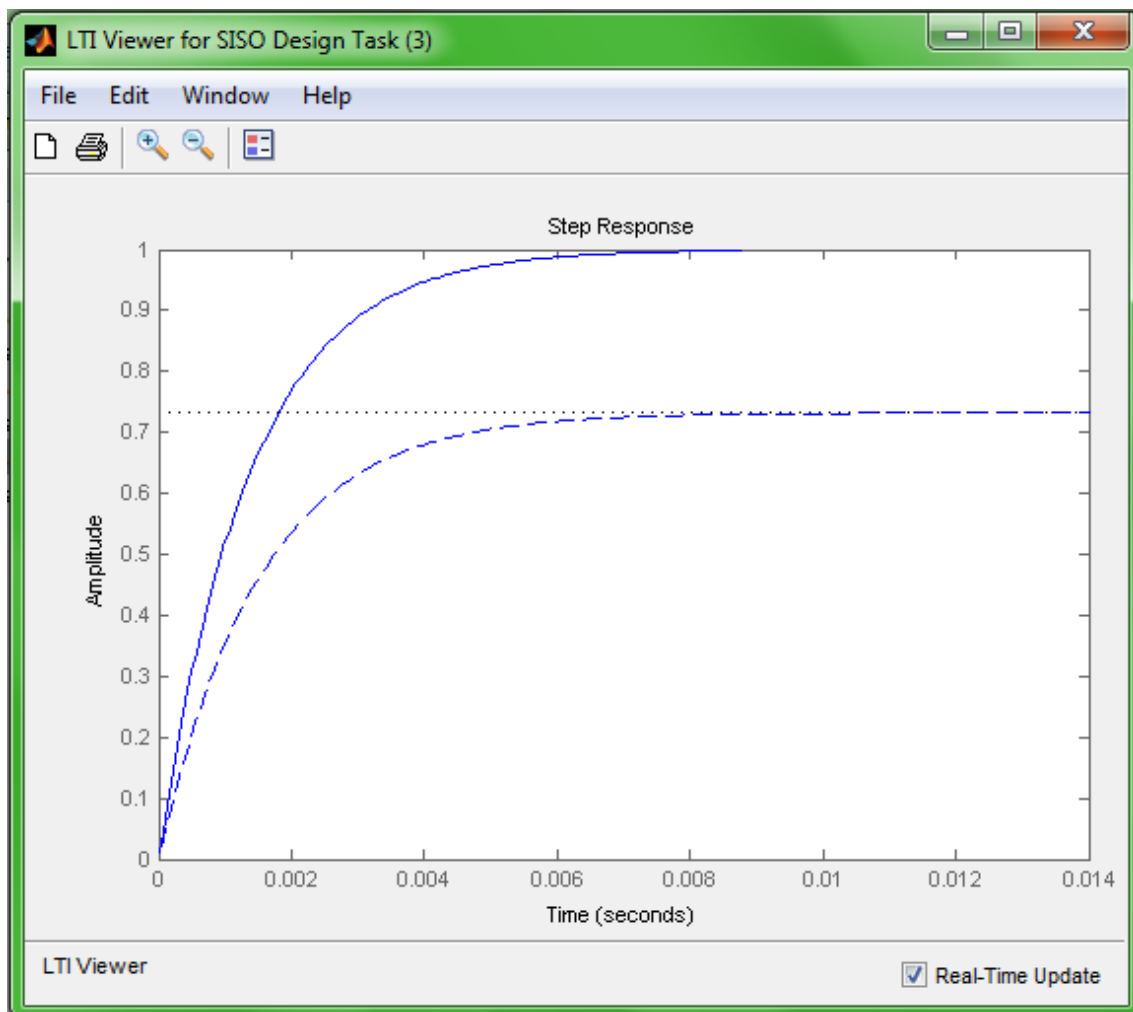
vyšel systém jako $F(p) = \frac{0,7255}{0,0032p+1}$, metodou nejmenších čtverců vyšel $F(p) = \frac{0,7309}{0,0015p+1}$

Navrhuje se spojitý PI regulátor, proto se v sisotoolu přidá pól regulátoru do nuly (integrator) a nula regulátoru se nastaví tak, aby se vykrátila s pólem soustavy, tedy $-\frac{1}{0,0015} = -660,8$, což se zaokrouhlí na -661, jak je vidět na obrázku Obr. 4.6.



Obr. 4.6: Ukázka nastavování parametrů regulátoru v sisotoolu, zdroj: vytvořeno autorem

Jakmile je nastavena nula, je z grafu frekvenční charakteristiky v sisotoolu vidět, že charakteristika má sklon -20 dB/dek v celém svém rozsahu a tudíž lze u regulátoru zvolit libovolné zesílení. Vhodné zesílení se určí za pomoci sisotoolu v záložce Analysis Plots, kde se u možnosti Plot 1 vybere možnost step a zaškrtnutím polí “Closed loop r to y” a “Plant G” se zobrazí přechodové charakteristiky soustavy s regulátorem a samotné soustavy. V grafu pak lze sledovat, jak se projeví zesílení regulátoru na přechodovou charakteristiku soustavy. Regulátor je třeba poměrně rychlý, zvolil jsem proto zesílení 1000. Výsledná přechodová charakteristika je na obrázku dole Obr. 4.7.



Obr. 4.7: Odezva na jednotkový skok soustavy s regulátorem (plná čára) a bez regulátoru (přerušovaná čára), zdroj: vytvořeno autorem

Výsledný přenos spojitého regulátoru tedy vyšel $F_R(p) = 1000 \frac{0,0015p+1}{p}$. V programu se ovšem pracuje s diskretním regulátorem. Z tohoto důvodu je třeba převést regulátor do tvaru, ze kterého jej lze snadno převést do diskretního:

$$F_R(p) = K_R(1 + T_D p + \frac{1}{T_I p}) \quad (4.2)$$

Vychází se z tvaru, který se vypočítal v sisotoolu.

$$F_R(p) = k_r \frac{T_r p + 1}{p} \quad (4.3)$$

Jednotlivé konstanty se upraví podle vztahů:

$$T_i = \frac{1}{k_r} \quad (4.4)$$

$$r_0 = \frac{T_r}{T_i} \quad (4.5)$$

$$r_i = \frac{1}{T_i} \quad (4.6)$$

Na tvar:

$$F_R(p) = r_0 + \frac{r_i}{p} \quad (4.7)$$

A dále pak podle rovnic:

$$T_I = \frac{r_0}{r_i} \quad (4.8)$$

$$K_R = r_0 \quad (4.9)$$

na požadovaný tvar

$$F_R(p) = K_R \left(1 + T_D p + \frac{1}{T_I p} \right) \quad (4.10)$$

kde derivační složka T_D je nulová a z tohoto důvodu se neuvažuje. [3] Pro rychlejší převod lze využít funkci “prevod“ v Matlabu. Vstupními argumenty funkce jsou k_r a T_r z prvního tvaru regulátoru a výstupem je K_R a T_I z tvaru požadovaného. Požadované hodnoty je dále třeba ještě přepočítat na diskrétní, podle vtaů:

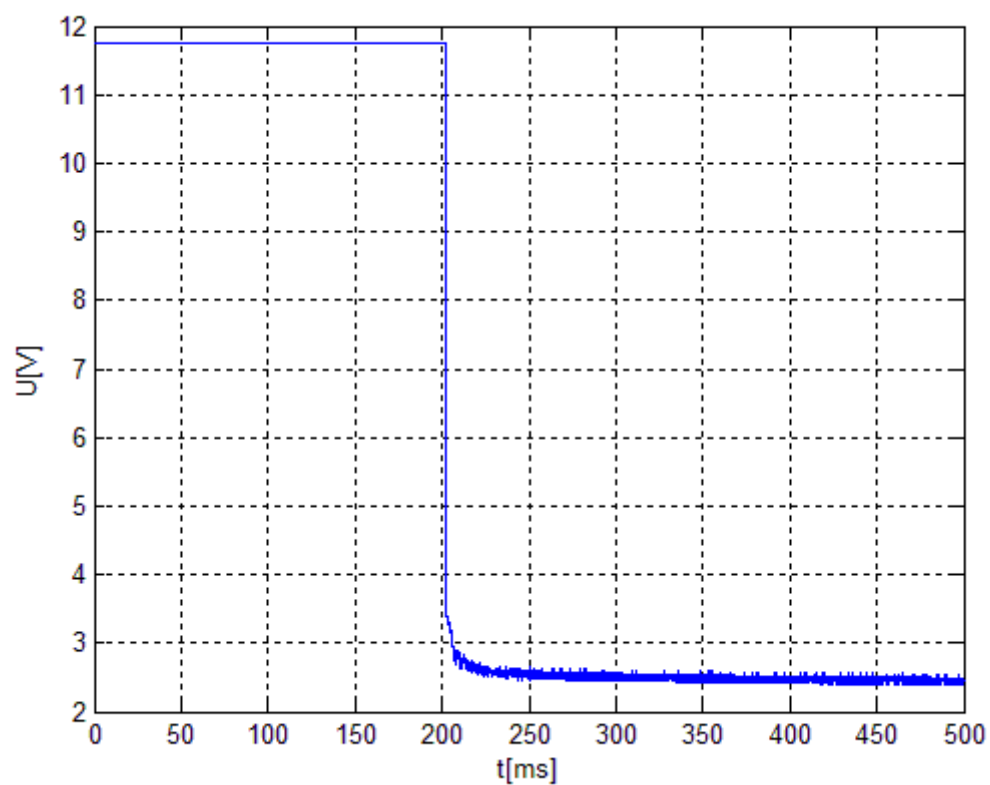
$$T_{IV} = K_R \frac{T_S}{T_I} \quad (4.11)$$

$$T_{DV} = K_R \frac{T_D}{T_S} \quad (4.12)$$

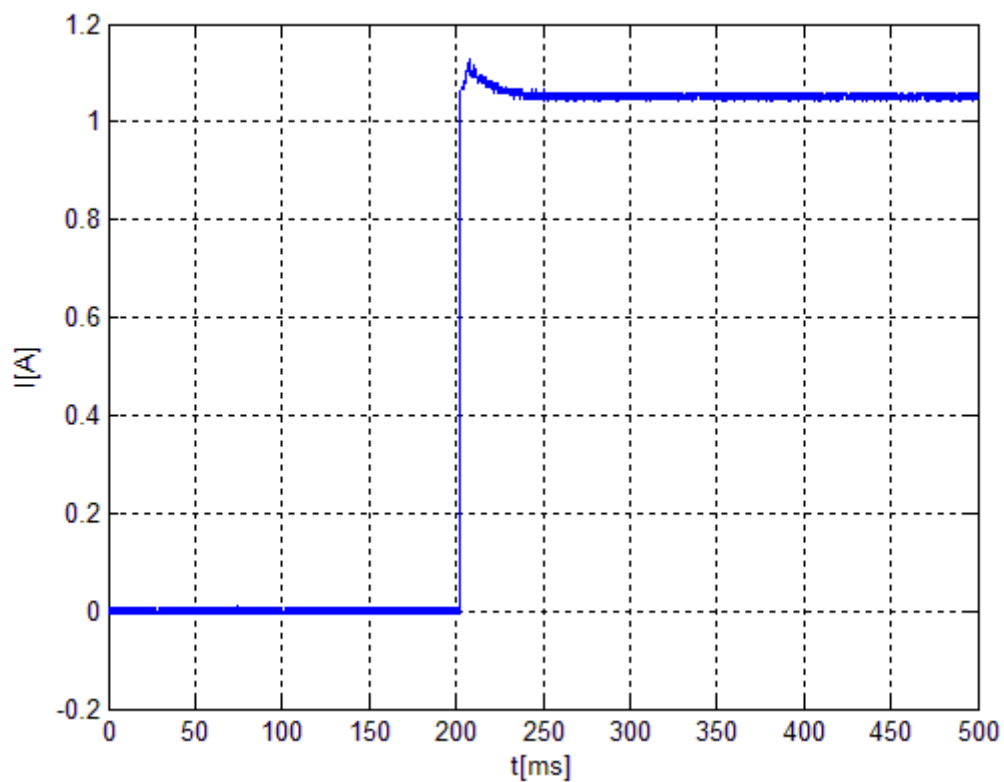
Kde T_{IV} je výsledná integrační časová konstanta a T_{DV} je výsledná derivační časová konstanta. K_R se nemění. Před zadáním regulátoru do programu je třeba ještě všechny konstanty vynásobit 2^8 , tedy 256, kvůli softwarovému provedení.

V tomto případě je tedy $K_R = 384$, $T_{IV} = 13$ a $T_D = 0$.

Pro identifikaci pomocí přechodové charakteristiky vyšel regulátor $K_R = 819$, $T_{IV} = 13$ a $T_D = 0$.



Obr. 4.8: Průběh napětí v regulované soustavě při zabrzděném motoru, zdroj: vytvořeno autorem



Obr. 4.9: Průběh proudu v regulované soustavě při zabrzděném motoru, zdroj: vytvořeno autorem

Výsledné přechodové charakteristiky zobrazují průběh proudu a napětí u soustavy s regulátorem identifikované pomocí přechodové charakteristiky. Pro změření byla použita funkce programu pro identifikování soustavy. Byla nastavena žádaná hodnota proudu (o něco vyšší než 1A) a v čase 200 ms byl motor spuštěn příkazem enable drive. Z grafu je patrné, že regulátor proudu funguje, a že omezil napětí na motoru v souladu se žádanou hodnotou proudu i při zablokovaném rotoru. Graf napětí ukazuje, že dokud motor nebyl spuštěn, regulátor nastavil maximální akční zásah a teprve po spuštění motoru začal správně regulovat.

4.4 Program v RT

Požadovanou hodnotu napětí lze v RT zadávat v rozsahu od -2000 do 2000 jako Duty Cycle. Tato hodnota je šířka pulsu, kde 2000 odpovídá střídě 100% a 0 odpovídá 0%. Znaménko pak určuje směr otáčení rotoru. Tato hodnota se na motoru projeví pouze v případě, že jsou vypnuty oba regulátory v FPGA. Jestliže je zapnutý regulátor proudu, změna této hodnoty nebude mít na rychlost motoru žádný vliv. Dále lze tlačítkem “Enable drive“ zapnout nebo vypnout motor a tlačítkem “Enable E-stop“ lze motor nouzově zastavit se zobrazením zprávy, že motor byl zastaven tímto tlačítkem. Všechny tři hodnoty se po bitech naskládají za sebe pomocí bloků Join numbers na jednu 64bitovou hodnotu. Přenos této hodnoty do FPGA je uskutečňován blokem FIFO Ultimate, což je DMA kanál přenášející 64 bitové integery.

V RT se v grafech zobrazují průběhy napětí, proudu a otáček v reálném čase. Grafy “Proud“ a “Napětí“ slouží pouze pro identifikaci pomocí přechodové charakteristiky a průběhy se v nich zanesou po stisknutí tlačítka “identifikovat“. Graf “Otáčky“ zaznamenává okamžitou hodnotu otáček motoru. Graf “Proud 2“ ukazuje průběh proudu v jednotkách 0.0062A, což je jednotka, ve které měří čidlo proudu. Tento graf je zde pro vizuální přehlednost, neboť žádaná hodnota proudu se zadává v těchto jednotkách v kolonce “Current setpoint“. Graf “Proud 3“ pak zobrazuje průběh proudu v Ampérech. Graf “Napětí 3“ zobrazuje okamžitou hodnotu napětí ve Voltech. V RT lze zadávat do příslušných kolonek (“Current PID Gains“ a “Velocity PID Gains“) velikosti jednotlivých složek regulátorů proudu a otáček.



Obr. 4.10: Uživatelské rozhraní RT, zdroj: vytvořeno autorem

Mimo hlavní smyčku jsou bloky pro připojení k FPGA a start obou DMA kanálů a jejich ukončení, protože start i ukončení DMA kanálů je třeba provést pouze jednou a jejich restart při každém průchodu smyčky by byl nežádoucí.

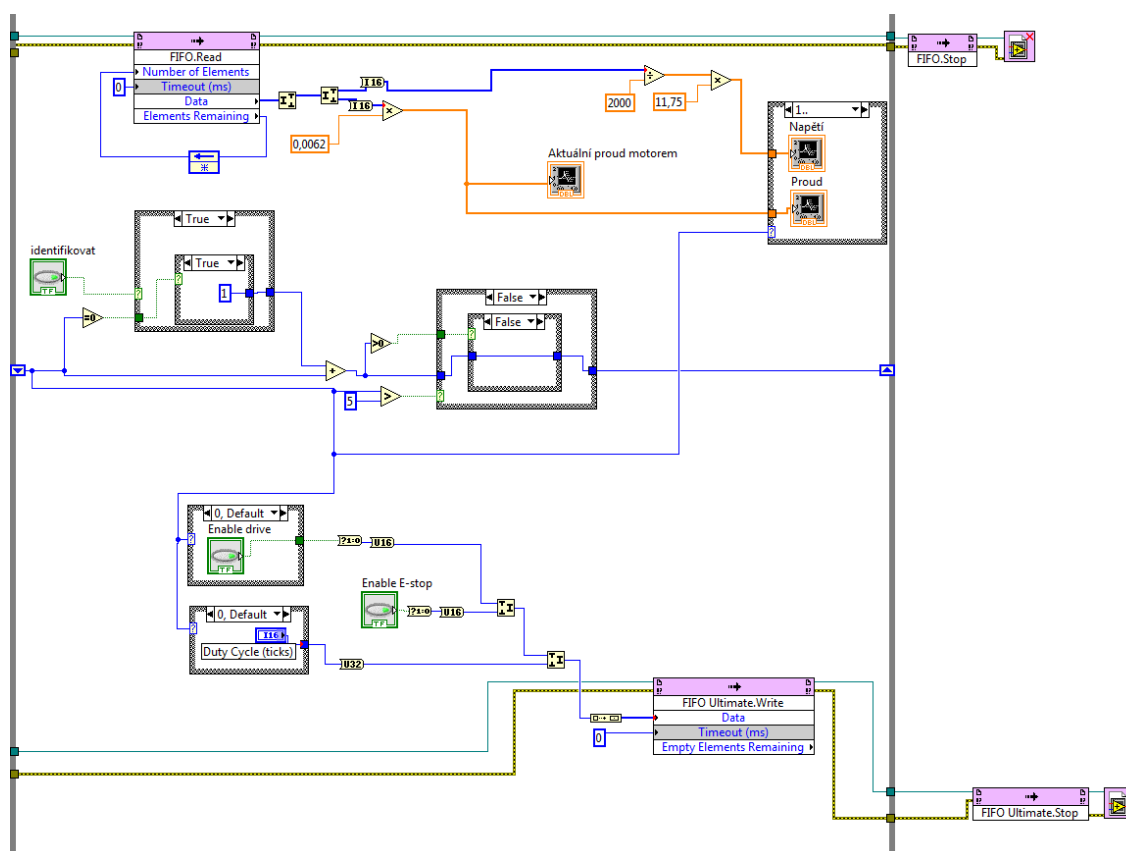
V horní části se nachází DMA kanál FIFO, který vyčítá údaje z FPGA jako 64bitovou proměnnou a dále těchto 64 bitů rozděluje na dvě čísla s polovičním počtem bitů blokem Split Numbers. Konečným výstupem z DMA kanálu jsou dvě 16bitové proměnné, kde jedna je aktuální napětí na motoru přepočítávané z PWM a druhá je aktuální proud motorem. Abychom dostali hodnotu proudu v ampérech, je třeba vynásobit hodnotu z bloku Current Sense konstantou 0,0062. Proud motorem se zobrazuje v bloku Chart s názvem "Aktuální proud motorem". Do dalších Chartů, označených jako "Napětí" a "Proud" vstupuje signál po dobu určenou časovačem, který se spouští tlačítkem "identifikovat" a to z toho důvodu, aby naměřenou frekvenční charakteristiku při identifikaci nepřemazala další data.

Tlačítko "identifikovat" je spínač, po jehož sepnutí se rozběhne vytvořený časovač. Za normálních okolností je hodnota na časovači nula a s každým dalším průchodem smyčky se k této hodnotě připočítává obsah dvou do sebe vnořených smyček Case. V případě sepnutí tlačítka "identifikovat" se smyčka Case přepne a druhá smyčka porovná, jestli je v hodnotě časovače nula. Pokud ano, do hodnoty časovače se přičte jednička. Tímto je zajištěno, že ve chvíli, kdy je časovač spuštěn, nelze opětovným stisknutím "identifikovat" ovlivnit dobu měření identifikace, která takto zůstává konstantní. Časovač dále vchází do dalších dvou vnořených smyček Case.

V případě, že do smyčky přijde nula, nic se s ní neděje. Jestliže přijde číslo větší než nula, při průchodu smyčkou Case se inkrementuje o 1. Jestliže přijde číslo větší než 5

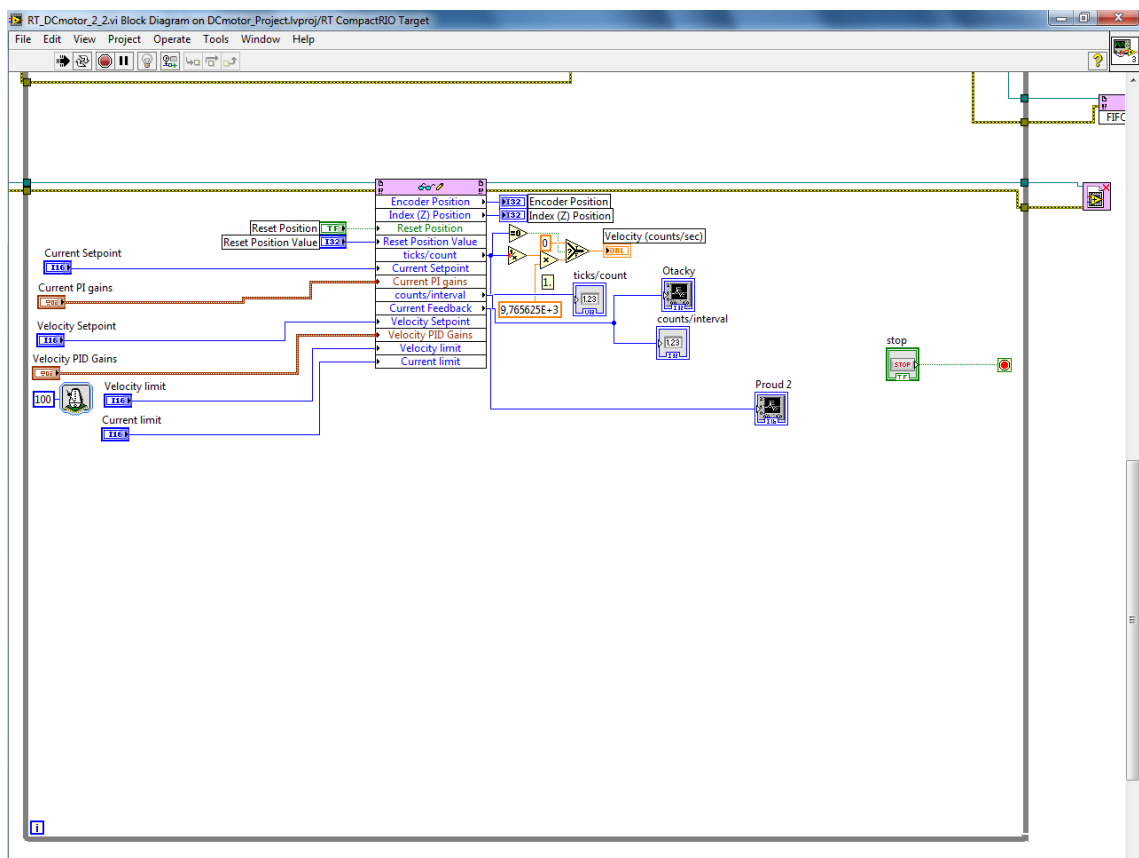
(aby byla získána celá přechodová charakteristika i ustálená hodnota proudu, bylo experimentálně určeno, že je třeba provádět měření přechodové charakteristiky po dobu 500 ms. Protože program v RT běží s periodou 100 ms, je třeba 5 průchodů smyčky), hodnota časovače se nastaví zpět na 0.

V době, kdy je na časovači nenulová hodnota, je dočasně vypnuta funkce tlačítka “Enable drive“, které je po celou dobu identifikace nastaveno na sepnuto, a také je ignorována hodnota v “Duty Cycle (ticks)“, kde se při hodnotě časovače 1 nastaví 0 a při hodnotách vyšších se nastaví 40% maximální hodnoty, tedy 800. Po skončení identifikace tyto tlačítka opět začnou fungovat.



Obr. 4.11: Výřez programu v RT, 1. část, zdroj: vytvořeno autorem

Ovládání regulátorů a enkodéru je přenášeno do FPGA pomocí bloku Read/Write Control Function. Tento blok přenáší lokální proměnné z RT do FPGA a opačně. Z RT do FPGA se posílá nastavení jednotlivých konstant proudového regulátoru “Current PI gains“, žádaná hodnota výstupního proudu “Current Setpoint“ a omezení maximální povolené hodnoty Setpointu “Current Limit“. Stejný přenos pak funguje u regulátoru otáček. Z FPGA do RT se posílají výstupní hodnoty enkodéru – pozice rotoru “Encoder Position“, a počet tiků enkodéru za jedno proběhnutí smyčky, což se dále přepočítává na rychlost v otáčkách za sekundu “Velocity (counts/sec)“.

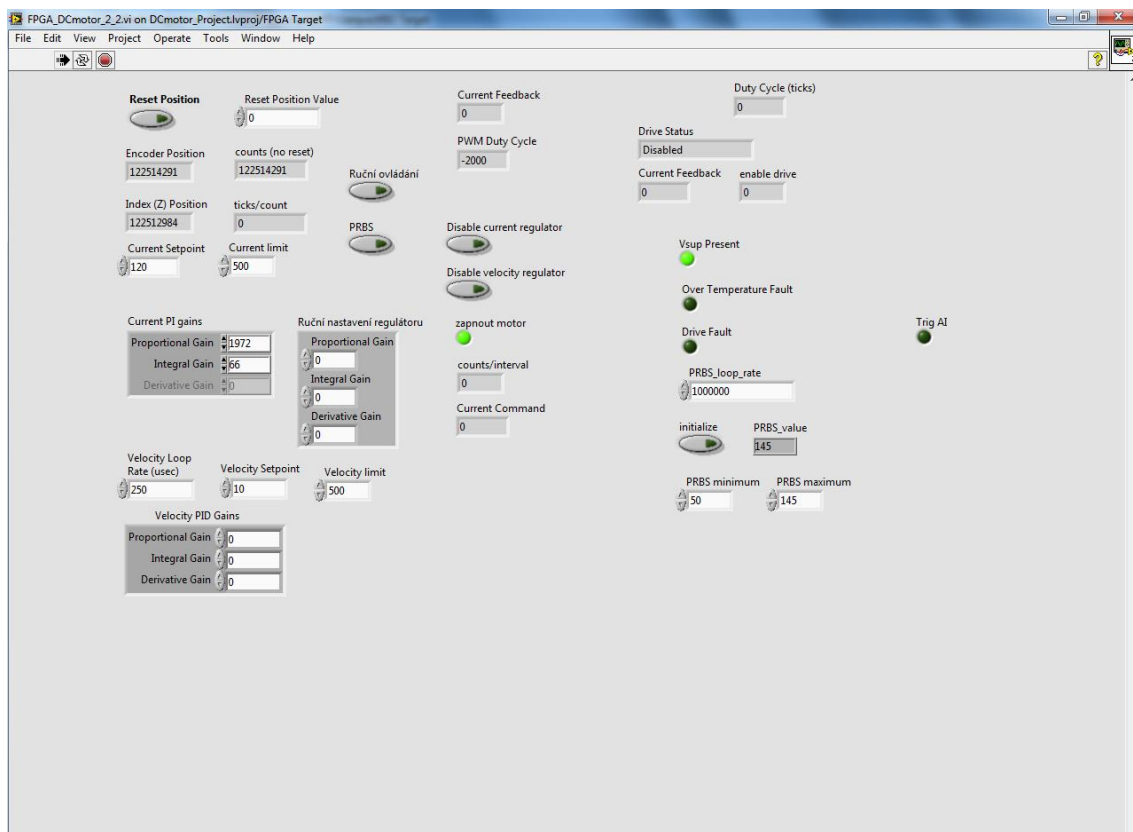


Obr. 4.12: Výřez programu v RT, 2. část, zdroj: vytvořeno autorem

4.5 Program v FPGA

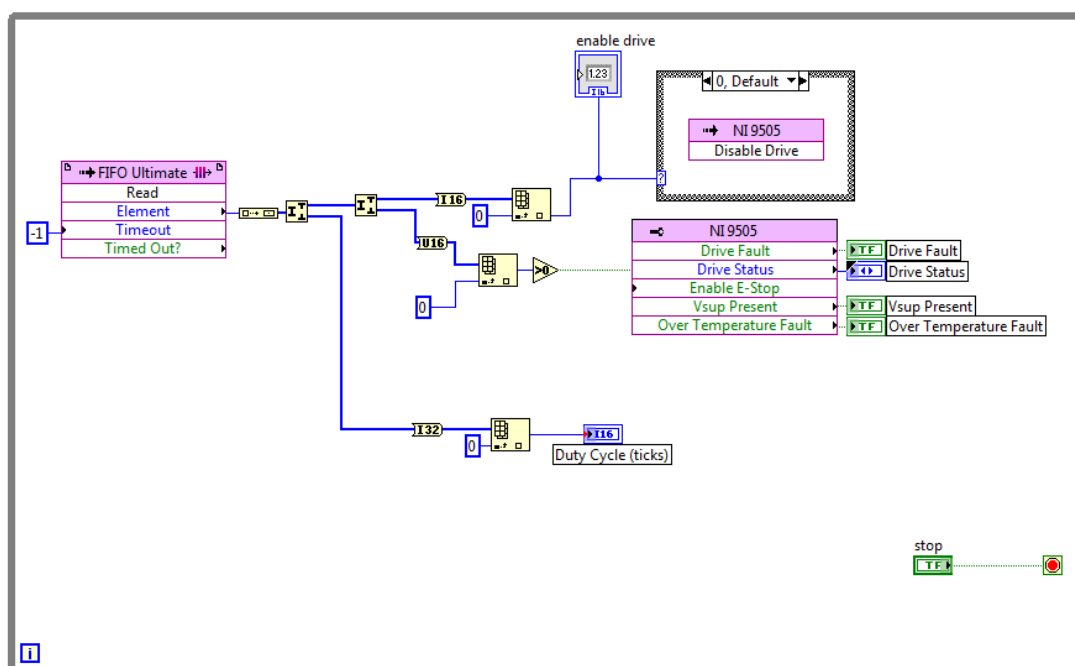
Program FPGA je tvořen několika smyčkami while, na rozdíl od RT. Důvodem je časování smyček a přehlednější rozdělení programu. V FPGA je několik tlačítek. Tlačítka “Disable Current Regulator” a “Disable Velocity Regulator” slouží pro vypnutí funkce proudového a otáčkového regulátoru, tlačítko “PRBS” přivede na vstup soustavy Pseudonáhodnou binární sekvenci, jejíž velikost lze nastavit v “PRBS maximum” a “PRBS minimum“. Je třeba také nastavit dobu posunutí PRBS o jednu hodnotu. Tato doba se nastavuje v μs .

V případě, že jsou vypnuty oba regulátory, bude PRBS hodnota napětí na vstupu motoru, pokud ale bude zapnut regulátor proudu, bude PRBS odpovídat žádané hodnotě proudu posílané do regulátoru. Této funkce lze využít v případě, že chceme zidentifikovat systém s regulátorem proudu pomocí MNČ. Tlačítko “initialize” zresetuje Smyčku PRBS, takže sekvence začne od začátku.



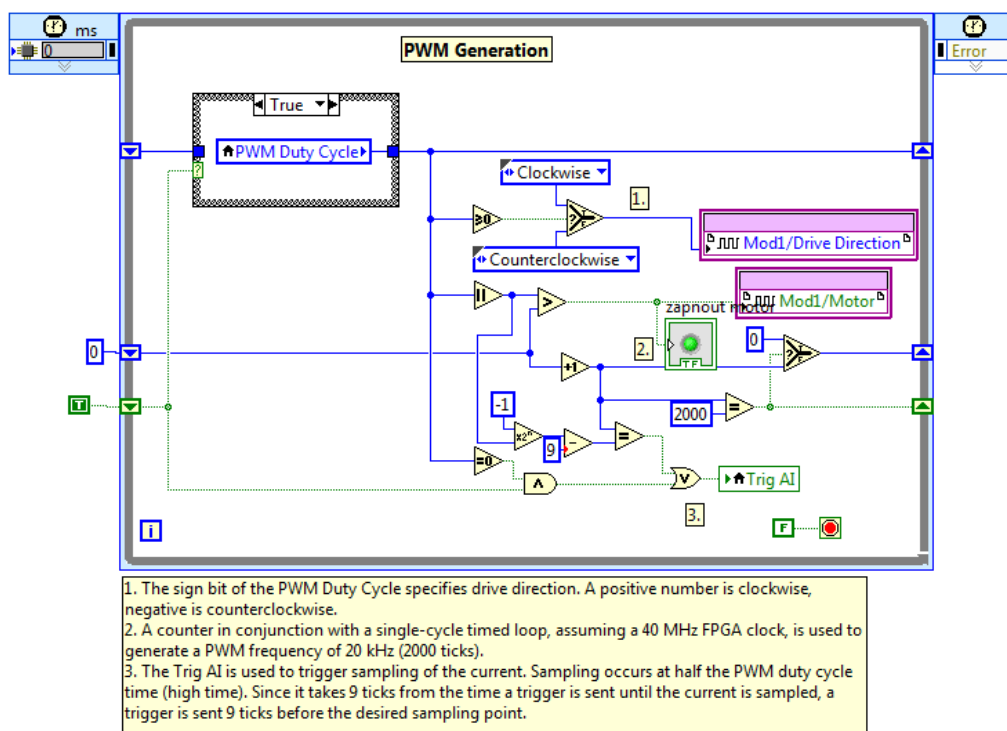
Obr. 4.13: Uživatelské rozhraní FPGA, zdroj: vytvořeno autorem

První smyčka vyčítá hodnoty z FIFO Ultimate a pomocí bloků Split Numbers rozdělí po bitech hodnoty z RT zpět do původního tvaru. DMA kanál posílá hodnotu ve formě pole, proto je ještě třeba použít bloky Index Array Function, které z pole vyberou jednu jeho hodnotu. Protože je v každém poli pouze jedna hodnota, stačí vyčíst první člen (s indexem 0). Hodnota “enable drive” je přivedena ke smyčce Case, ve které je v případě, že přijde nula blok FPGA I/O Method Node s funkcí Disable Drive a v případě, že je hodnota “enable drive” jedna, nachází se ve smyčce tento blok s funkcí Enable Drive. Hodnota “Enable E-Stop” je přivedena do bloku FPGA I/O Property Node, který je nakonfigurován na tuto funkci a zároveň tento blok zobrazuje stav motoru pomocí tří diodových indikátorů (“Vstup Present”, “Over Temperature Fault” a “Drive Fault”) v uživatelském rozhraní FPGA a také zobrazuje, zda je motor zapnutý nebo vypnutý v indikátoru “Drive Status”. Hodnota “Duty Cycle (ticks)” je přes lokální proměnnou poslána do další smyčky, kde se vyčítá pouze v případě, že je vypnutý regulátor proudu.



Obr. 4.14: Smyčka while v FPGA pro zapnutí a vypnutí motoru, zdroj: vytvořeno autorem

Smyčka “PWM Generation“ obsahuje lokální proměnnou “PWM Duty Cycle“, která je výstupem regulátoru. Podle jejího znaménka se za pomoci porovnávacího bloku Select Function určuje směr otáčení rotoru. Kladné znaménko a nula znamenají otáčení rotoru ve směru hodinových ručiček, záporné znaménko znamená otáčení v protisměru. Dále je zde triggerování následující smyčky, kde se vyčítá proud motorem. Toto triggerování je vytvářeno lokální proměnnou “Trig AI“ a nastává v polovině PWM cyklu. Toho je docíleno pomocným čítačem od 0 do 2000, který využívá posuvného registru k ukládání aktuální hodnoty a poté se vynulovává pomocí bloku Select Function, kdy v případě, že je jeho hodnota 2000, přepne blok jeho hodnotu na 0. Protože od doby, kdy se pošle trigger do doby, než je proud vzorkován uběhne čas, který odpovídá 9 tikům procesoru, je triggerování posláno o 9 tiků napřed.



Obr. 4.15: Smyčka "PWM Generation", zdroj: vytvořeno autorem

Další smyčka while v sobě obsahuje Flat sequence Structure, což je struktura, která zajišťuje sekvenční vykonávání příkazů. Nejprve se tedy vykonají příkazy v okně úplně vlevo a poté se postupuje do dalších oken. V prvním okně je triggerování celého cyklu. Toto triggerování pochází ze smyčky "PWM Generation" a zajišťuje správnou funkci vzorkování proudu pro regulátor.

Ve druhém okně je vyčítání aktuálního proudu pomocí bloku I/O Node nastaveného na Mod1/Current Sense. Jeho hodnota je ukládána do lokální proměnné "Current Feedback". V tomto okně je také Loop timer, který nastavuje periodu smyčky na 50 μ s, z důvodu dostatečného vzorkování proudu.

Třetí okno obsahuje několik do sebe vnořených struktur Case, které se ovládají tlačítky "Disable Current regulator", "Disable Velocity Regulator", "PRBS" a "Ruční ovládání".

První je tlačítko "PRBS". Jeho stisknutí přivede PRBS signál na vstup motoru, v případě, že je vypnut regulátor proudu, a nebo na vstup regulátoru proudu.

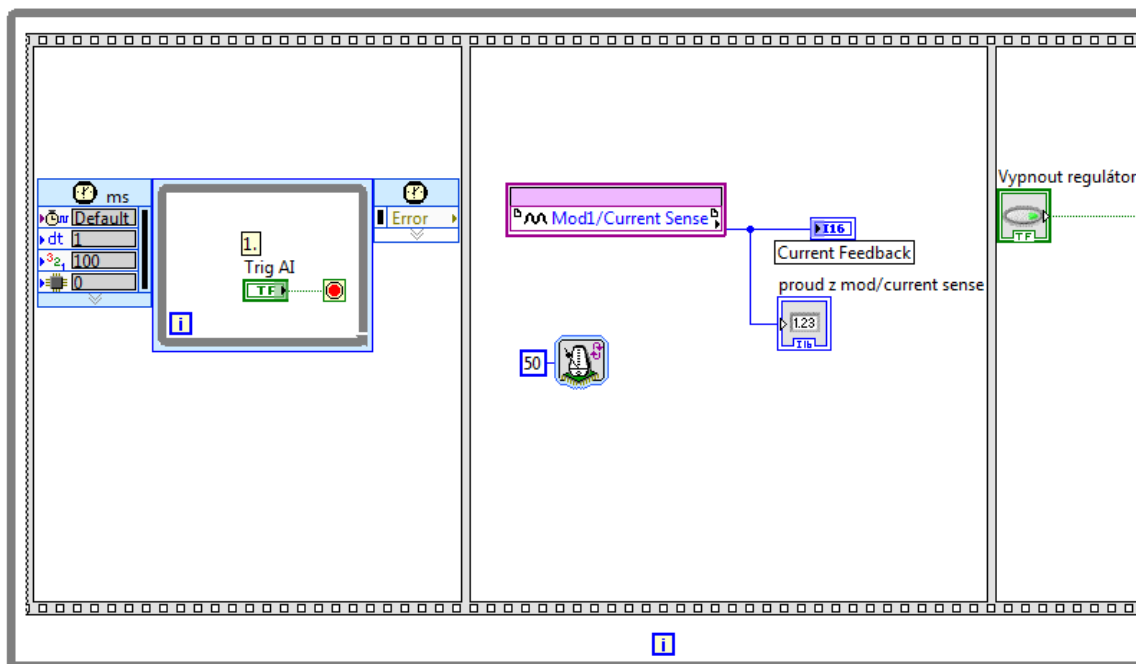
Tlačítko "Disable Velocity Regulator" ovládá regulátor otáček. V případě, že je sepnuté, je vstupem proudového regulátoru hodnota "Current setpoint", tedy uživatelem nastavená žádaná hodnota proudu. Pokud toto tlačítko seplé není, je vstupem regulátoru proudu hodnota "Current Command", což je výstup regulátoru otáček.

Dalším tlačítkem je "Disable Current Regulator". Po jeho sepnutí je ve smyčce pouze vyčítání z lokální proměnné "Duty Cycle (ticks)", tedy přímý přenos žádané hodnoty napětí zadané obsluhou v RT. Jestliže je regulátor vypnutý a je zapnuta PRBS, je vstupem motoru hodnota signálu PRBS. V případě, že je regulátor zapnutý, nachází se ve smyčce samotný PID regulátor ve formě SubVI Node, tedy podprogram FPGA.

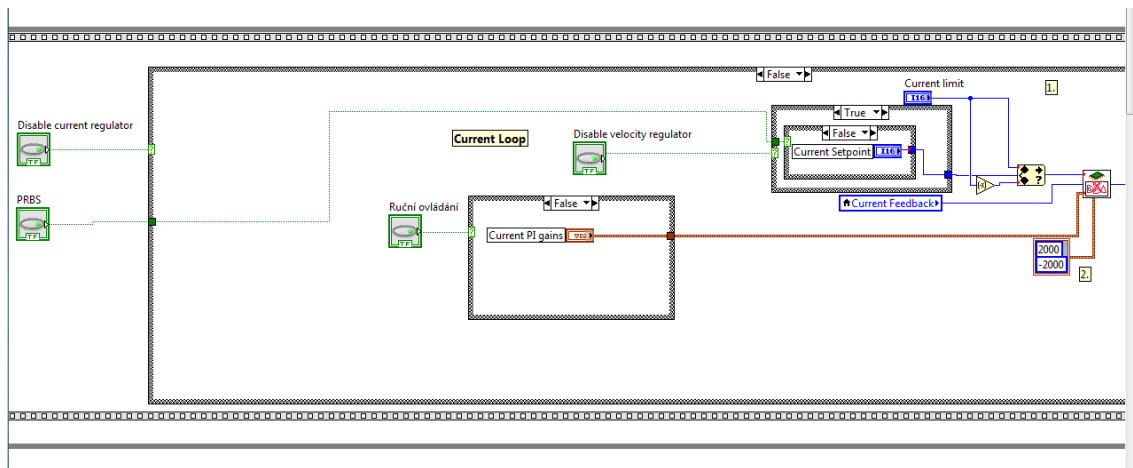
Je zde také tlačítko “Ruční ovládání“, které bylo vytvořeno pro účely ladění programu. Tlačítko ovládá nastavení konstant regulátoru, po jeho stisknutí je možné v uživatelském rozhraní FPGA zadávat konstanty ručně. Pokud je tlačítko vypnuté, konstanty se do FPGA posílají automaticky z RT.

Vstupem regulátoru je žádaná hodnota “Current Setpoint“ nebo “Current Command“ (podle toho, zda je zapnut regulátor otáček), které jsou blokem In Range and Coerce Function omezeny hodnotou “Current Limit“. (blok zjistí, zda hodnota “Current Setpoint“ náleží do intervalu omezeného hodnotou $\pm$ “Current Limit“ a pokud ne, nastaví hodnotu + nebo - “Current Limit“ podle toho, jestli byl “Current Setpoint“ vyšší nebo nižší. Jestliže se hodnota “Current Setpoint“ do tohoto intervalu vejde, je přivedena na výstup bloku) Do regulátoru dále vstupuje hodnota proudu z předchozího okna “Current Feedback“, která je tímto o jeden krok zpožděná. Dalším vstupem do regulátoru je “Output Range“, což je rozsah výstupu, který je nastaven od -2000 do 2000, aby odpovídal PWM. Posledním vstupem pak jsou vypočítané hodnoty konstant regulátoru, které je nutné vynásobit 256. Výstup regulátoru pak vede přímo do dalšího okna.

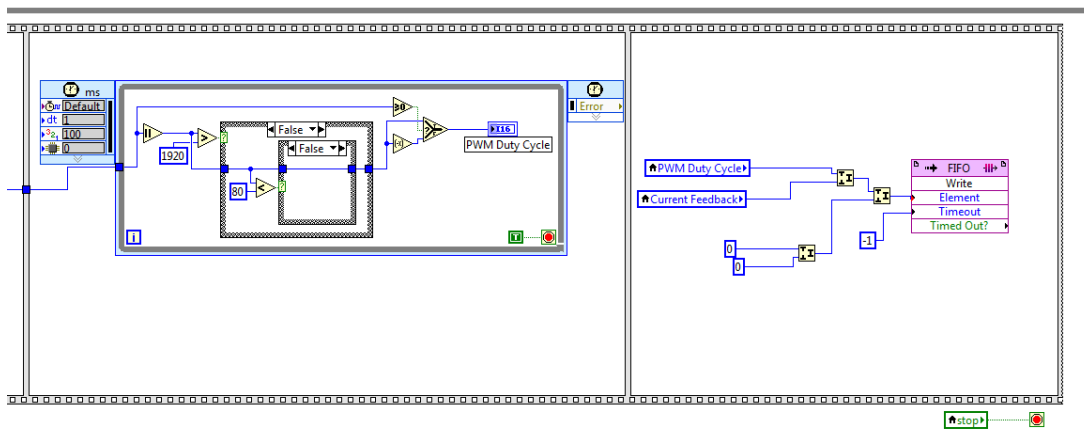
Ve čtvrtém okně se nachází pouze omezení výstupu regulátoru. V případě, že je hodnota výstupu vyšší než 1920, je upravena na 2000 a v případě, že je menší než 80, je upravena na 0 (u záporných hodnot se výstup upravuje zrcadlově, -1920 na -2000 a -80 na 0). Je to z toho důvodu, že minimální šířka pulsu PWM generovaného kontrolérem je 2 μ s. Při přepínání o frekvenci 2000kHz je proto použitelný rozsah PWM od 4% do 96%, tedy od 80 do 1920. Kontrolér ale umí generovat i pulsy o střídě 0% a 100%, a proto se hodnoty nespadající do rozsahu 80-1920 upravují. Po této operaci se hodnota výstupu ukládá do proměnné “PWM Duty Cycle“, která je v posledním okně společně s hodnotou “Current Feedback“ posílána zpět do RT DMA kanálem “FIFO“, kde jsou obě hodnoty zobrazovány do grafu při identifikaci.



Obr. 4.16: Smyčka pro regulaci proudu, 1. část, zdroj: vytvořeno autorem



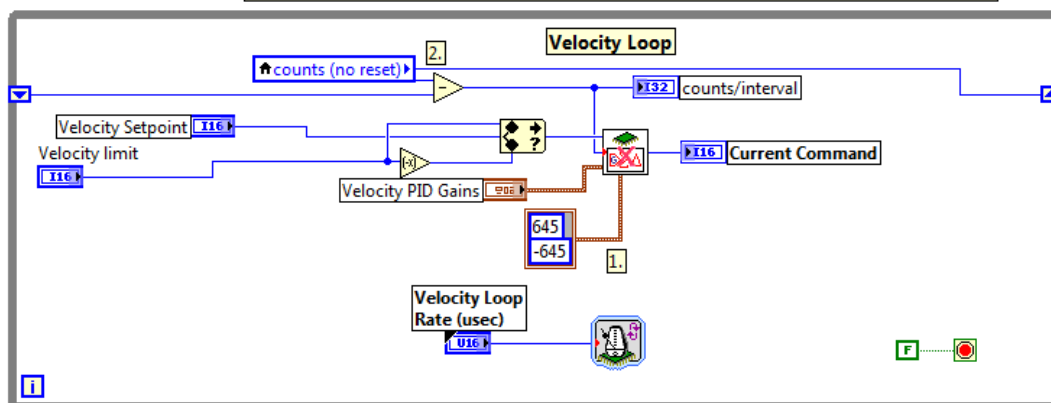
Obr. 4.17: Smyčka pro regulaci proudu, 2. část, zdroj: vytvořeno autorem



Obr. 4.18: Smyčka pro regulaci proudu, 3. část, zdroj: vytvořeno autorem

Regulátor otáček je umístěn ve smyčce “Velocity loop“. Jeho vstupní uspořádání je totožné s regulátorem proudu, hodnotu aktuálních otáček vyčítá z enkodéru pomocí proměnné “Counts (no reset)“. S touto hodnotou pracuje tak, že si uloží její hodnotu z předcházející smyčky pomocí posuvného registru a v každém kroku pak odečítá počet otáček v minulém kroku od počtu otáček v aktuálním kroku. Tímto je schopen zjistit počet otáček v dané smyčce. Výstupem regulátoru je proměnná “Current Command“, tedy žádaná hodnota proudu, která je následně přivedena na vstup regulátoru proudu.

This loop is used to control the velocity defined by Velocity Setpoint. The output, Current Command, feeds the Current Loop. The loop consists of a closed loop PID, with the feedback and coming from the Encoder Loop.
 1. The output range of the PID loop is set to min/max current capability of +/-12A.
 2. Calculate the velocity as counts/loop. Each time the velocity loop executes, the previous counts (no reset) value is subtracted from the current one.



Obr. 4.19: Smyčka "Velocity Loop"

Smyčka “Encoder Loop“ v sobě obsahuje enkodér. Ze snímače vycházejí tři různé signály, které mají tvar pulzů – Phase A, Phase B a Index. Pulzy A a B jsou vůči sobě mírně posunuté a jeden tak vždy do enkodéru přichází o něco dříve než druhý. Z tohoto údaje se vyčítá směr otáčení motoru. Jestliže přijde pulz A dříve než B, motor se točí proti směru hodinových ručiček, jestliže přijde první pulz B, motor se točí po směru. Pulz Index je referenční hodnota a generuje se vždy, když se rotor natočí do základní polohy, tedy jednou za otáčku.

The screenshot shows a LabVIEW block diagram for a position loop control system. The diagram is set to a 1 ms time scale. It includes a 'Read the current encoder phase states' block, an 'Encoder Loop' block, a 'Count Case' block, and various signal processing blocks like 'Non reset counts', 'Reset Position Value', 'Reset Position', 'Encoder position', 'Tick count at last encoder count', and 'Time between encoder counts'. The diagram is set to a 1 ms time scale and includes a 'Error' indicator.

1. "counts (no reset)" is used to facilitate calculation of velocity in FPGA. For example, it can be used to calculate counts/loop of a position loop without erroneous values caused by "Reset Position".

2. "ticks/count" is used in RT to calculate velocity by utilizing the accuracy of the FPGA system clock.

3. Count Case (False Case) - If no quadrature counts are detected for >0.5 seconds, "ticks/count" is set to 0 to indicate a velocity of 0. Assumes an FPGA system clock rate of 40MHz.

[illegible][illegible]

Funkcia „prumer“ je určená pre validáciu dát. Má 10 vstupných parametrov (5 a 5) a 2

nutné, aby vstupem byla přechodová charakteristika ve tvaru náběžné hrany. Jako vstup se zadá 5 průběhů proudu a 5 průběhů napětí, výstupem je jeden průběh proudu a jeden průběh napětí, které jsou aritmetickým průměrem 5 vstupních charakteristik. (Funkce nejprve najde počátek náběžné hrany a poté všechny průběhy upraví tak, aby počátek všech pěti náběžných hran byl v jednom bodě. Výsledkem je pak průběh omezený nejkratší naměřenou charakteristikou.)

Funkce "identifikace" slouží pro identifikaci za pomoci přechodové charakteristiky, která se získá při měření se zastavenou hřídelí (Locked rotor test). Jejím výstupem jsou hodnoty K a T, kde K je zesílení soustavy a T je časová konstanta. Jako vstup této funkce je vhodné použít data získané z funkce "prumer".

Funkce "MNC" a "MNC3" jsou téměř totožné, slouží pro identifikaci systému metodou nejmenších čtverců. funkce "MNC" aproximuje vstupní charakteristiky modelem prvního řádu, funkce "MNC3" pak modelem třetího řádu. Diskrétní modely jsou poté převedeny na spojitě pomocí funkce matlabu d2c. Funkce "MNC", "MNC3" a "identifikace" lze použít i pro data, která nebyla naměřena v programu, který byl vytvořen k této bakalářské práci, je v nich ale nutno změnit periodu vzorkování Ts na hodnotu vzorkovaného signálu.

Funkce "prevod" slouží pro rychlé převedení spojitěho regulátoru, navrženého v sisotoolu, na diskrétní tvar. Funkce se skládá pouze ze za sebou jdoucích rovnic (4.4) až (4.9). Zdrojové kódy všech funkcí jsou uvedeny v příloze A.

5 ZÁVĚR

V této bakalářské práci byl vytvořen program, sloužící k identifikaci stejnosměrného motoru. Motor byl identifikován Metodou nejmenších čtverců a za pomoci přechodové charakteristiky. Podle identifikovaných parametrů byly poté navrženy a naprogramovány PI regulátory proudu pro tento motor metodou frekvenčních charakteristik. Regulátory byly zároveň prakticky vyzkoušeny ve vytvořeném programu a byla dokázána jejich funkčnost.

V rámci práce bylo také vytvořeno několik funkcí v Matlabu, které slouží pro identifikaci naměřených přechodných charakteristik a pro usnadnění přepočítávání konstant regulátoru.

Program je plně funkční, obsahuje nastavitelný regulátor proudu a otáček a nástroje pro naměření průběhů akčních veličin.

Motor byl identifikován Metodou nejmenších čtverců, kde jeho přenos vyšel $F(p) = \frac{0,7309}{0,0015p+1}$ a pomoci přechodové charakteristiky, kde jeho přenos vyšel $F(p) = \frac{0,7255}{0,0032p+1}$. Drobné odchylky v přenosech byly způsobeny pravděpodobně proměnným odporem kartáčů v motoru, kdy natočení hřídele ovlivňuje velikost vnitřního odporu a tím i proud procházející motorem.

V rámci práce byla vytvořená laboratorní úloha pro tento motor, kde si studenti mohou vyzkoušet identifikaci reálného motoru a návrh vhodného regulátoru pro něj.

Literatura:

- [1] *What Is NI CompactRIO?* [online]. c2014 [cit. 2015-01-05]. Dostupné na URL: <<http://www.ni.com/compactrio/whatis/>>
- [2] *[TUT] [C] Newbie's Guide to AVR PWM (Incomplete)* [online]. c2008 [cit. 2015-01-03]. Dostupné na URL: <<http://www.avrfreaks.net/forum/tut-c-newbies-guide-avr-pwm-incomplete>>
- [3] Bláha, P., Vavřín, P.: *Řízení a Regulace I*. Brno: c2015 VUT Brno: VUT Brno, 2013, 215s.
- [4] *Using DMA FIFO to Develop High-Speed Data Acquisition Applications for Reconfigurable I/O Device* [online]. c2014 [cit. 2015-01-08]. Dostupné na URL: <<http://www.ni.com/white-paper/4534/en/>>
- [5] ĎAĎO, Stanislav. *Senzory a měřicí obvody*. 1. vyd. Praha: ČVUT, 1996, 315 s. ISBN 80-010-1500-9.
- [6] VLACH, Jaroslav, Josef HAVLÍČEK a Martin VLACH. *Začínáme s LabVIEW*. 1. vyd. Ilustrace Viktorie Vlachová. Praha: BEN - technická literatura, 2008, 247 s. ISBN 978-80-7300-245-9.
- [7] NEBORÁK, J. *Modelování a simulace elektrických regulovaných pohonů*. VŠB - Technická univerzita Ostrava, 2002.
- [8] *MID – Modelování a Identifikace* [online]. c2015 [cit. 2015-05-10]. Dostupné na URL: <<https://sites.google.com/site/modelovaniaidentifikace/cviceni-5---vstupni-signaly-pro-identifikaci>>