

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky a mezioborových inženýrských studií



DIPLOMOVÁ PRÁCE

*Malý laboratorní servopohon pro výuku
automatického řízení*

Liberec 2006

Josef Loos

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií



Studijní program: M2612 – Elektrotechnika a informatika

Studijní obor: 3902T005 – Automatické řízení a inženýrská informatika

Malý laboratorní servopohon pro výuku automatického řízení

A Small Laboratory Servo-drive for Teaching Automatic Control

Diplomová práce

Autor:

Josef Loos

Vedoucí a konzultant diplomové práce:

Dr. Ing. Mgr. Jaroslav Hlava

Datum: 19.05.2006

TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky a mezioborových inženýrských studií

Katedra řídicí techniky

Akademický rok: 2004/2005

ZADÁNÍ DIPLOMOVÉ PRÁCE

pro: **Josefa Loose**

studijním programem: M 2612 – Elektrotechnika a informatika

obor: 3902T005 – Automatické řízení a inženýrská informatika

Vedoucí katedry Vám ve smyslu zákona o vysokých školách č.111/1998 Sb. určuje tuto diplomovou práci:

Název tématu: **Malý laboratorní servopohon pro výuku automatického řízení**

Zásady pro vypracování:

1. Stávající částečně dohotovený model servopohonu doplňte o měření proudu procházejícího kotvou motoru, tak aby bylo možné do regulační struktury zařadit podřazený regulátor proudu.
2. Signály z IRC snímače lze v prostředí Simulinku a Control Webu vyhodnotit jen obtížně a navíc je použitelnost modelu výrazně omezena, neboť naprostá většina běžných laboratorních karet nemá IRC vstupy. Proto navrhnete a postavte obvod (na bázi vhodně zvoleného jednočipového mikropočítače doplněného o D/A převodník), který vyhodnotí signál z IRC snímače a na výstupech poskytne údaj o rychlosti a poloze v podobě analogového napětíového signálu. K naprogramování mikropočítače použijte podle vlastní volby assembler, nebo C.
3. Navrhnete a realizujete programové vybavení pro řízení a vizualizaci servopohonu v prostředí Control Web 5.
4. Vytvořte pedagogickou dokumentaci k servopohonu obsahující jeho popis, matematický model a návrh regulačních experimentů zaměřených především na následující dva tématické okruhy: kaskádní řízení, časově optimální řízení.

Rozsah grafických prací: dle potřeby dokumentace

Rozsah průvodní zprávy: cca 40 až 50 stran

Seznam odborné literatury:

- [1] Chyský, J., Novák, J., & Novák, L. (1998) Elektronické aplikace ve strojírenství: Mikroprocesory. Praha: Vydavatelství ČVUT
- [5] Punčochář, J. (1999), Operační zesilovače v elektronice, Praha: BEN – technická literatura
- [6] Razím, M., & Horáček, P. (1985), Optimální a adaptivní řízení, Praha: Vydavatelství ČVUT
- [7] Souček, P. (1997), Pohony výrobních zařízení – servomechanismy, Praha: Vydavatelství ČVUT
- [8] Suchánek, V. (1982), Silnoproudá elektrotechnika v automatizaci. Praha: SNTL

Vedoucí diplomové práce: Dr. Ing. Mgr. Jaroslav Hlava

Konzultant:

Zadání diplomové práce: **20. 10. 2004**

Termín odevzdání diplomové práce: **20. 5. 2005**

L.S.

.....
Ing. Libor Tůma, Csc.

Vedoucí katedry

.....
Doc.Dr.Ing. Jiří Maryška, Csc.

Děkan

V Liberci dne: 19. 5. 2006

Téma: Malý laboratorní servopohon pro výuku automatického řízení

Anotace:

Hlavním cílem této diplomové práce je sestavení elektronického ovládání servopohonu a následné regulační školní experimenty.

V úvodní části je popsán model, který byl vytvořen v ročníkovém projektu. Model se skládá ze dvou stejnosměrných elektromotorků, setrvačníku a inkrementálního snímače.

Ve druhé části je popis zaměřen na návrh a sestavení elektronických obvodů pro snímání inkrementálních signálů z IRC snímače, dále pak obvod pro kaskádní řízení, který udává elektrické napětí úměrné proudu kotvou motoru.

Ve třetí části je popsán návrh a realizace programového vybavení pro řízení a vizualizaci servopohonu v softwarovém prostředí Control Web 5.

Čtvrtá část uvádí pedagogickou dokumentaci k modelu servopohonu, která obsahuje jeho popis, matematický model, obrazový přenos a návrh regulačních experimentů.

Theme: A Small Laboratory Servo-drive For Teaching Automatic Control

Annotation:

The aim of this diploma thesis is the realization of the electronic control of the servo drive and the following regulation experiments at schoolwork.

In the first part the model created during the year project is described in detail. The model consists of two DC electric motors, a balance-wheel and an incremental sensor.

The suggestion and the realization of the electrical network for scanning of the incremental signals from the IRC sensor are presented in the second part. The electrical network for the cascade control which indicates the voltage adequate to the current of the motor armature is also presented in the second part of the diploma thesis.

Part three deals with the description of the suggestion and the realization of the software equipment for control and visualization of the servo drive in the software environment of Control Web 5.

In the part four the educational documentation for the model of the servo drive is introduced. It contains the description of the servo drive, its mathematical model, the image transmission and the suggestion for the regulation experiments.

Prohlášení:

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

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

Jsem si vědom 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).

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně pod vedením vedoucího diplomové práce a použil jsem pouze podklady (literaturu, projekty, software atd.) uvedené v příloženém seznamu.

V Liberci dne:

.....
Josef Loos

Poděkování:

Na tomto místě bych chtěl poděkovat vedoucímu mé diplomové práce Dr. Ing. Mgr. Jaroslavu Hlavovi za jeho ochotu, pomoc a odborných připomínek při zpracování mé práce, bez kterých by nemohla vzniknout.

Můj vděk také patří všem, kteří mě při práci podporovali.

Obsah:

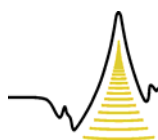
Obsah	7
Seznam obrázků	10
Seznam tabulek	12
1 Úvod	13
2 Celková hardwarová koncepce návrhu modelu	15
2.1 Vyhodnocení signálů z IRC snímače – snímání polohy, otáček	15
2.1.1 Inkrementální IRC snímač MOZ30	15
2.1.2 Realizace modelu – výběr elektronických součástek	16
2.1.2.1 Dekodér / Čítač HCTL 2016	16
2.1.2.1.1 Digitální šumový filtr	16
2.1.2.1.2 Kvadrurní dekodovací logika	17
2.1.2.1.3 Načasování rozhraní dekodéru IRC signálů	17
2.1.2.2 Jednočipový mikropočítač	19
2.1.2.2.1 Požadavky na mikropočítač	19
2.1.2.2.2 Výběr a popis jednočipového mikropočítače – AT89S51	19
2.1.2.2.3 Výběr krystalu pro mikropočítač AT89S51	21
2.1.2.2.4 Resetovací obvod - TL7705	22
2.1.2.2.5 Podpůrné obvody mikroprocesoru	22
2.1.2.3 D/A převodníky	23
2.1.2.3.1 D/A převodník (Digitálně/Analogový) - MAX 502	23
2.1.2.4 DC/DC měnič TEM3 – 521	24
2.1.2.5 Zdroj referenčního napětí LM336	25
2.1.3 Závěr návrhu snímání a vyhodnocení signálů z IRC snímače	25
2.1.4 Softwarová realizace mikropočítače	27
2.1.4.1 Nastavení časovače	27
2.1.4.2 Čtení dat z převodníku	27
2.1.4.3 Stanovení úhlové polohy hřídele stejnosměrného motoru	28
2.1.4.4 Snímání rychlosti stejnosměrného motoru	29
2.1.4.4.1 Nadeřinování rychlostních konstant v assembleru	29
2.1.4.4.2 Stanovení rychlosti	29
2.1.4.4.3 Směr otáčení	30

2.1.4.4.4	Měření rychlosti	31
2.1.4.4.5	Znaménková korekce dat	32
2.1.4.4.6	Zápis rychlosti na D/A převodník	33
2.2	Snímání proudu v kotvě ss. motoru	33
2.3	Převod polarity výstupního analogového signálu z PCI-1711	34
2.4	Chlazení výkonového operačního zesilovače	35
2.5	Elektrodynamická brzda	36
2.6	Vizuální a mechanické zpracování servopohonu	37
2.6.1	Popis mechanického zapojení	37
3	Popis zapojení úlohy a signálů	38
3.1	Úvod	38
3.1.1	Popis komponent na obrázku (Obr. 3.2)	38
3.1.2	Popis signálů soustavy	39
3.2	Reálné měření na modelu servopohonu v prostředí MATLAB	
	Simulink – RealTime Toolbox	40
3.2.1	Odezvy výstupních signálů	41
3.2.1.1	Přechodová charakteristika reálného modelu	41
3.2.1.2	Snímání polohy hřídele reálného modelu	41
3.2.1.3	Vliv poruchy na rychlost otáčení hřídele modelu	42
3.2.1.4	Snímání parametrů při reverzaci servopohonu	43
4	Vizualizace ovládání modelu v Control Web 5	44
4.1	Úvod	44
4.2	Popis návrhu ovládacího panelu	45
4.3	Připojení signálů modelu na kanály karty	49
4.4	Měření na servosystému	49
5	Matematický popis, model a obrazový přenos modelu	51
5.1	Úvod	51
5.1.1	Odpor kotvy stejnosměrného motoru	51
5.1.2	Konstrukční konstanta stejnosměrného motoru	52
5.1.3	Moment setrvačnosti modelu	52
5.1.3.1	Moment setrvačnosti setrvačníku bez hřídelí	53
5.1.3.2	Moment setrvačnosti hřídelky setrvačníku	53
5.1.3.3	Celkový moment setrvačnosti setrvačníku	53

5.2	Matematický popis modelu	54
5.2.1	Simulace modelu pomocí matematického popisu	57
5.3	Obrazový přenos modelu	58
5.3.1	Identifikace modelu pomocí obrazového přenosu	59
5.4	Ověření identifikovaného systému	60
6	Regulační experimenty modelu pomocí obrazových přenosů	61
6.1	Úvod regulace	61
6.1.1	Obrazový přenos regulátoru	62
6.2	Optimální časové seřízení PID regulátoru	62
6.2.1	Průběh optimálního řízení na reálném modelu	65
6.3	Kaskádní regulace	66
7	Závěr	69
	Literatura	71
	Seznam použitých zkratk	72
A	Příloha	73
A.1	Stejnoseměrný elektromotor s permanentními magnety P2MO476	73
A.1.1	Technické schéma ss. motoru (rozměrové parametry)	73
A.1.2	Charakteristiky ss. motoru	74
A.2	Optoelektronický generátor impulsů: MOZ30	75
A.2.1	Elektronické data snímače	75
A.2.2	Mechanické, ostatní data snímače MOZ30	76
A.3	Elektronická schéma	77
A.3.1	Elektronické schéma snímání IRC signálu	77
A.3.1.1	Seznam použitých elektronických součástek	78
A.3.1.2	Deska plošného spoje snímací jednotky	79
A.3.2	Celkové schéma elektronického zapojení úlohy	80
A.3.2.1	Seznam použitých elektronických součástek	81
A.4	Výpis programů pro optimalizaci PID regulátoru	82

Seznam obrázků:

- 1.1 Mechanické zapojení modelu
- 1.2 Blokové schéma návrhu snímání a vyhodnocení IRC signálů
- 2.1 Princip inkrementálního senzoru
- 2.2 HCTL 2016
- 2.3 Čtyř kvadrurní rozlišení signálů A, B
- 2.4 Načasování rozhraní HCTL 2016
- 2.5 Elektronické zapojení obvodu HCTL 2016
- 2.6 Zapojení vývodů mikroprocesoru AT89S51
- 2.7 Načasování čtecího cyklu obvodu HCTL 2016
- 2.8 Časování procesoru
- 2.9 Obvod TL7705
- 2.10 Zapojení resetovacího obvodu TL7705
- 2.11 D/A převodník MAX 502
- 2.12 Zapojení D/A převodníku MAX502
- 2.13 Vývody DC/DC měniče
- 2.14 Vývody LM336
- 2.15 Elektrické zapojení obvodu LM336
- 2.16 Blokové schéma zpracování IRC signálu
- 2.17 Rozdělení 16-ti bitového datového slova
- 2.18 Napěťový údaj o poloze
- 2.19 Nastavení Carry bitu při kladném a záporném směru otáčení
- 2.20 Snímání proudu v kotvě pomocí OZ a malého rezistoru
- 2.21 Zapojení předzesilovače
- 2.22 Elektronické schéma zapojení teplotního spínače
- 2.23 Zapojení spínané zátěže, pomocí TTL výstupů z analog. karty
- 2.24 Mechanické zapojení servopohonu
- 3.1 PC karta PCI-1711
- 3.2 Technologické schéma
- 3.3 Blokové schéma řízení modelu servopohonu
- 3.4 Snímání rychlosti otáčení servopohonu
- 3.5 Snímání natočení hřídele servopohonu



- 3.6 Snímání vlivu poruchy na servopohonu
- 3.7 Zobrazení dat při reverzaci modelu
- 4.1 Ovládací panel servopohonu
- 4.2 Otevření archivace
- 4.3 DBase viewer
- 4.4 InCalc viewer
- 4.5 Data viewer
- 4.6 Grafické znázornění snímaných signálů v Control Web5
- 5.1 Schematické znázornění modelu
- 5.2 Schéma rozměrů setrvačníku
- 5.3 Blokové schéma modelu
- 5.4 Závislost rychlosti na čase t
- 5.5 Závislost otáček na čase t
- 5.6 Závislost proudu na čase t
- 5.7 Závislost momentu na čase t
- 5.8 Závislost polohy na čase t
- 5.9 Závislost výkonu na čase t
- 5.10 Blokové schéma obrazových přenosů
- 5.11 Model dynamického systému servopohonu
- 5.12 Graf rychlosti v závislosti na čase t
- 5.13 Graf otáček v závislosti na čase t
- 5.14 Ověření identifikovaného systému
- 6.1 Zpětnovazební řízení servopohonu
- 6.2 Optimalizace parametrů PID regulátoru
- 6.3 Model pro výpočet kritéria (soubor pidkra.mdl)
- 6.4 Model pro porovnání regulačních pochodů (soubor pidsima.mdl)
- 6.5 Regulované veličiny $y_o(t)$, $y_p(t)$
- 6.6 Schéma kaskádní regulace
- 6.7 Blokové znázornění kaskádního řízení servopohonu
- 6.8 Průběh kaskádní regulace
- A.1 Stejnoseměrný motor P2MO476
- A.2 Technické schéma stejnosměrného motoru
- A.3 Charakteristiky stejnosměrného servomotoru

- A.4 IRC snímač MOZ30
- A.5 Rozměrové parametry snímače MOZ30
- A.6 Elektronické obvodové schéma snímání IRC signálu
- A.7 Deska plošného spoje snímací jednotky
- A.8 Osazení plošného spoje snímací jednotky
- A.9 Celkové obvodové elektronické schéma úlohy

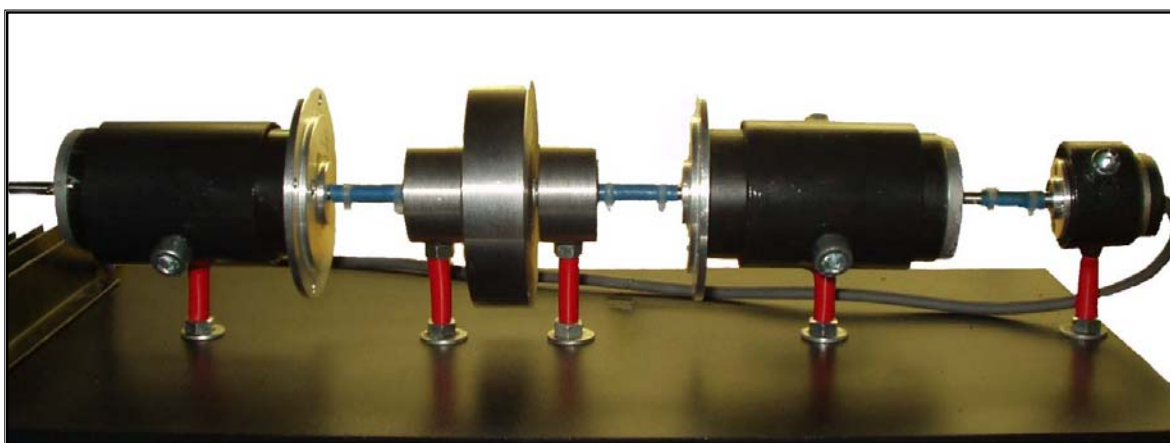
Seznam tabulek:

- 2.1 Popis vývodů obvodu HCTL 2016
- 2.2 Všeobecné parametry procesoru AT89S51
- 2.3 Popis základních vývodů procesoru AT89S51
- 2.4 Popis portů procesoru AT89S51
- 2.5 Rozšířený popis portu 3
- 2.6 Popis vývodů D/A převodníku MAX 502
- 2.7 Pravdivostní tabulka signálů /WR, /CS
- 2.8 Elektrické parametry DC/DC měniče
- 3.1 Nastavení signálů v MATLAB SIMULINKu
- 4.1 Zapojení signálů modelu
- 4.2 Měřené údaje servopohonu
- 5.1 Popis obrázku (Obr. 5.1)
- A.1 Elektronické data snímače MOZ30
- A.2 Ostatní data snímače MOZ30
- A.3 Mechanické data snímače MOZ30

1 Úvod

Jelikož je v současné době automatické řízení a regulace dosti časté téma, byl vytvořen v ročníkovém projektu pro účel výuky školní laboratorní model servopohonu. Pro kvalitní podmínky regulace je nutné využít maximální informace o mechanismu. Jednoduše řečeno, využijeme rychlosti otáček a proudu v kotvě stejnosměrného motoru ve zpětné vazbě regulačního řízení.

Cílem této diplomové práce je tedy navázat a dokončit model servomechanismu, který by bylo možné ovládat počítačem. Hlavní částí modelu tvoří navzájem hřídelově propojeny dva identické stejnosměrné elektromotory (P2MO476), setrvačnick a inkrementální snímač (MOZ30) viz obrázek (Obr. 1.1). Jeden z motorků je buzen napěťovým signálem, je tedy hnací, na kterém je pomocí zmíněného inkrementálního čidla snímána poloha a rychlost otáčení hřídele.

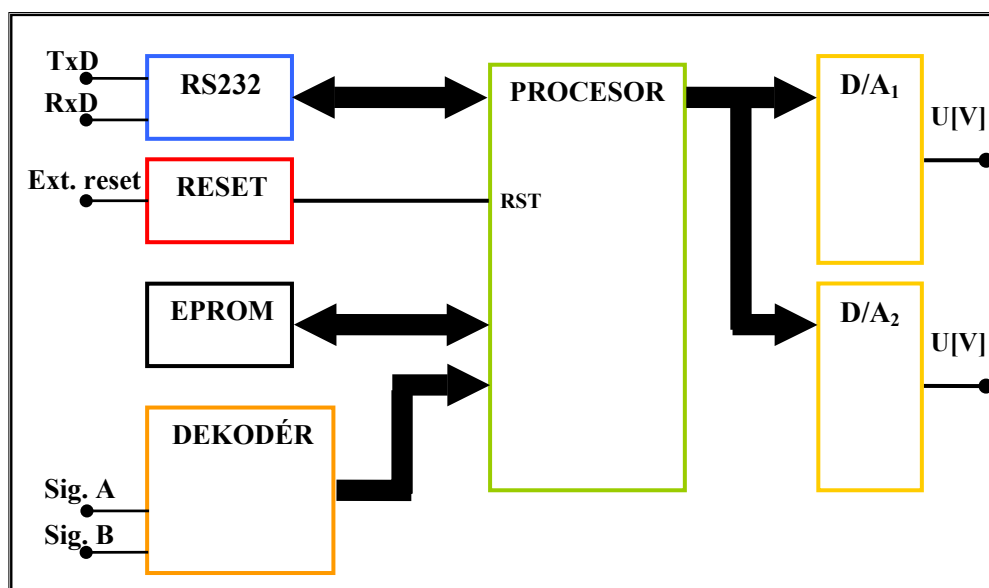


Obr. 1.1. Mechanické zapojení modelu

Druhý stejnosměrný elektromotor tvoří elektrodynamickou brzdu, kde odběr proudu vyrobeného se maří na teplo v odpornících viz kapitola 2.5. Tím je možné vytvářet brzdny moment hnacího elektromotoru a simulovat na něm poruchu.

V ročníkovém projektu se ukázalo, že vyhodnocení signálů z IRC snímače v softwaru MATLAB SIMULINK není tak docela jednoduché. Navíc je použitelnost modelu omezena tím, že nepracuje s jinou PC laboratorní kartou než-li MF 604, neboť běžné karty nemají IRC vstupy. Vývoj se tedy zabývá především návrhem a sestavením elektronického obvodu viz obrázek (Obr.1.2). Ten bude vyhodnocovat signály z IRC

snímače pomocí jednočipového mikropočítače, doplněného o D/A převodníky, tak aby na jednom bylo analogové napětí úměrné rychlosti a na druhém poloze. Jelikož se v regulační kaskádě stejnosměrného motoru obvykle na nejnižším stupni pracuje s regulací proudu, je vhodné zařadit do stávajícího modelu obvod, který bude udávat napětí úměrné proudu kotvy hnacího ss. motoru. Tím vznikne vstupně-výstupní analogové univerzální zařízení připojitelné k libovolné laboratorní PC kartě.



Obr. 1.2. Blokové schéma návrhu snímání a vyhodnocení IRC signálů

Další částí diplomové práce je připojení modelu k počítači pomocí multifunkční vstupně-výstupní karty Adventech **PCI-1711**, která bude zajišťovat měření a řízení signálů z prostředí programu MATLAB Simulink.

Na kapitulu o řízení navazuje další část práce obsahující popis a tvorbu vizualizace řízení servopohonu pomocí softwaru Control Web 5.

V závěru práce je rozebrána problematika vytvoření matematického popisu a obrazového přenosu modelu. Její výsledky jsou využity pro návrh několika variant řízení.

2 Celková hardwarová koncepce návrhu modelu

2.1 Vyhodnocení signálů z IRC snímače – snímání polohy, otáček

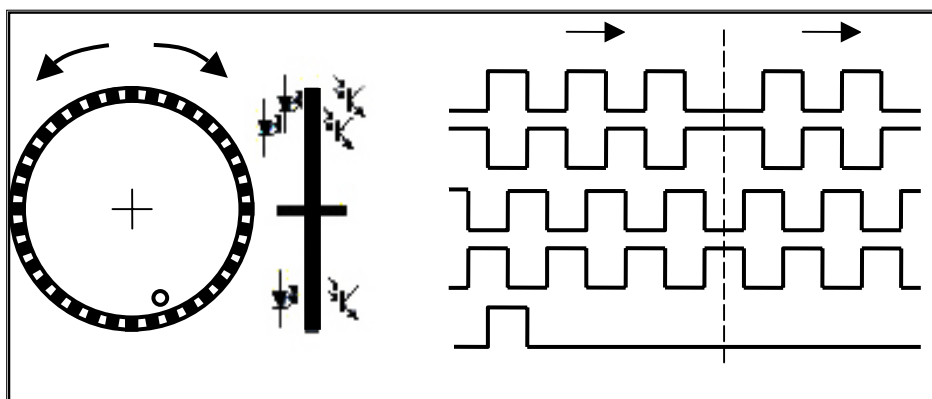
Tato kapitola popisuje návrh a samostatnou realizaci řešení vyhodnocení signálů z inkrementálního rotačního snímače.

2.1.1 Inkrementální IRC snímač MOZ30

Základem navrženého a vytvořeného systému je inkrementální rotační snímač MOZ30. Název inkrementální charakterizuje princip činnosti založený na otáčivém mezikruží s pravidelně se střídajícími průhlednými a neprůhlednými ploškami (ryskami), které při otáčení přerušují emitované světlo svítivé diody (LED) umístěné na jedné straně tohoto mezikruží. Do optické cesty mezi zdrojem a přijímačem světla je zařazen ještě nepohyblivý maskovací kotouč s ryskami o stejné rozteči, jako má kotouč pohyblivý. Při rozlišení smyslu otáčení, musí být maskovací kotouč senzoru polohy opatřen druhým segmentem s ryskami posunutými vůči ryskám prvního segmentu o úhel

$$\alpha = \left(k + \frac{1}{2}\right) \frac{2\pi}{n}, \quad (1.1)$$

kde n je počet rysek na obvodu kotouče a k je celé číslo. Signál z prvního fotosenzoru se označuje A, signál z druhého fotosenzoru B. Detekováním změny fáze signálů A a B získáme informaci o změně směru otáčení (Obr. 2.1). Otáčivý kotouč má i doplňkový otvor a je doplněn dalším zdrojem světla a fotosenzorem detekující tzv. výchozí (referenční) pozici v podobě jednoho pulsu na otáčku. Tento signál I se označuje jako nulový, nebo referenční.

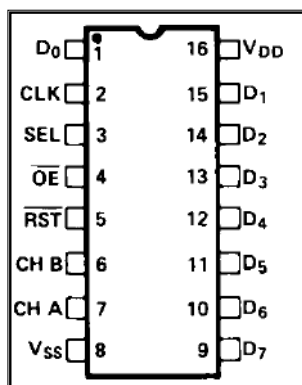


Obr. 2.1. Princip inkrementálního senzoru

Mechanické a elektronické specifikace inkrementálního senzoru MOZ30 jsou uvedeny v příloze A.2.

2.1.2 Realizace modelu – výběr elektronických součástek

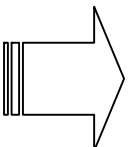
2.1.2.1 Dekodér / Čítač HCTL 2016



Obr. 2.2. HCTL 2016

Analogový obvod dekodér-čítač HCTL 2016 je určený pro dekódování kvadraturních signálů z IRC senzorů. Tento obvod obsahuje digitální filtr, čtyř kvadraturní dekódovací logiku, binární obousměrný (up / down) čítač a osmibitové sběrníkové rozhraní. Pomocí Schmittova klopného obvodu s CMOS vstupy je vstupní šum filtrován a umožňuje tak bezpečnou operaci v rušivých prostředích.

Tab. 2.1. Popis vývodů HCTL 2016

Pin	Označení	Funkce
1	D0	8bitový výstupní port (bit 0)
2	CLK	Vstup pro externí hodinový signál
3	SEL	Vstup povolující 8bitový výstup bufferu
4	/OE	Vstup pro přepínání Low a High bajtu
5	/RST	Vstup pro resetování čítače
6	CHB	Vstup pro připojení B signálu z IRC snímače
7	CHA	Vstup pro připojení A signálu z IRC snímače
8	Vss	Napěťová zem
9	D7	 8bitový výstupní port (bit 1-7)
10	D6	
11	D5	
12	D4	
13	D3	
14	D2	
15	D1	
16	Vdd	Napájecí napětí (+5 V)

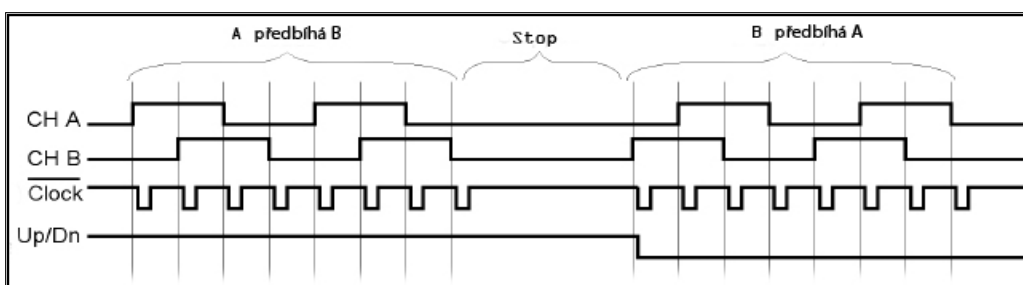
2.1.2.1.1 Digitální šumový filtr

Vstupní digitální filtr zavrhuje přicházející šum z kvadraturních signálů. K realizování zamítnutí šumu užívá vstupní oblast dvě techniky. Jednak pomocí užití Schmittova klopného obvodu, nebo pomocí zpoždění tří cyklových taktů. Spojením těchto dvou technik jsou zamítnuty malé napěťové šумы a velké trvalé šumové špičky,

které se typicky vyskytují u systémů motorových aplikací.

2.1.2.1.2 Kvadrurní dekódovací logika

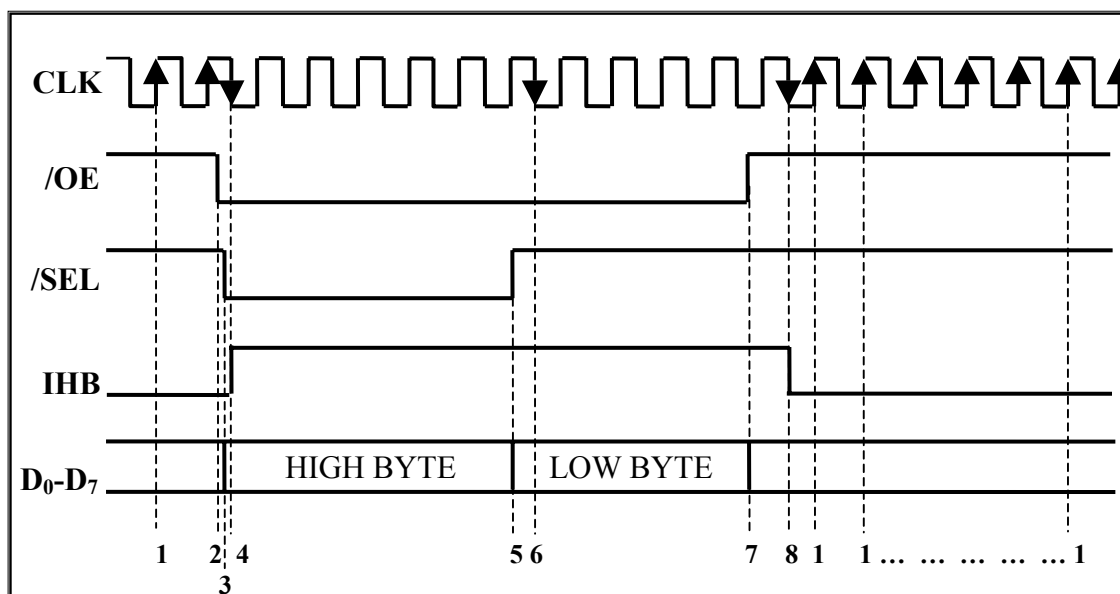
Dekódovací systém násobí rozlišení vstupních signálů A, B faktorem čtyři (4×dekódování). Výstupem dekodéru je čítací signál (/Clock) a směrový signál (Up/Dn). Obrázek (Obr. 2.3) představuje čtyř kvadrurní dekódování, kde detekované hrany jsou čelo a týl obou signálů. Předbíhá-li signál A signál B, je provedeno čítání nahoru (Up), naopak je provedeno čítání dolů (Dn).



Obr. 2.3. Čtyř kvadrurní rozlišení signálů A, B

Pro správnou funkčnost obvodu HCTL 2016 je nutné nastavit časové rozhraní, které udává obrázek (Obr. 2.4).

2.1.2.1.3 Načasování rozhraní dekodéru IRC signálů

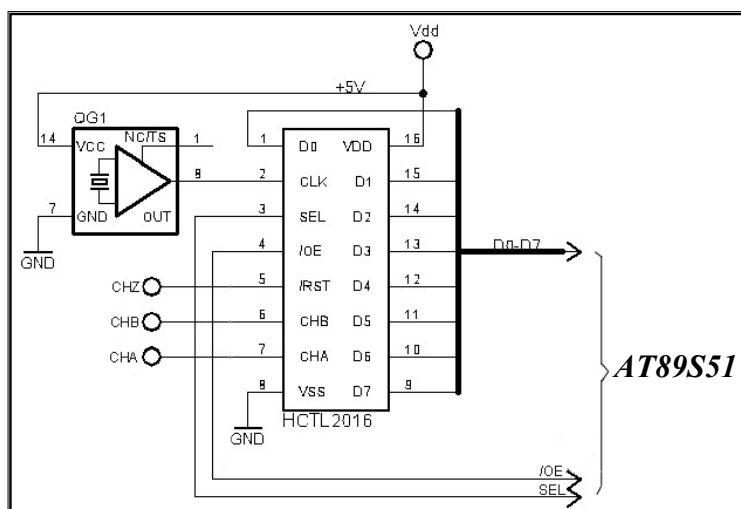


Obr. 2.4. Načasování rozhraní HCTL2016

Popis obrázku 2.4:

1. Na náběžnou hranu hodin signálu **CLK** se čítaná data přemísťují do pozic datových latch registrů a bránicí signál **IHB** má hodnotu logické nuly (low).
2. Je-li řídicí signál **/OE** nastaven na logickou nulu (low), jsou povoleny výstupy z multiplexoru.
3. Jestliže hodnota řídicího signálu **/SEL** je v logické nule (low), pak je povolen příjem vyššího pořadí datového slova (HIGH BYTE).
4. Dojde-li během sestupné hrany hodin **CLK** k logické 0 na signálech **/OE** a **/SEL**, vnitřní signál **IHB** je aktivovaný. Ten blokuje nová data přenesené do čítače.
5. Jde-li hodnota signálu **/SEL** do logické jedničky (high), výstupní data se mění z vyššího bajtu (HIGH BYTE) na nízký bajt (LOW BYTE).
6. Zde nastává první ze dvou resetovacích podmínek pro bránicí logiku a to, když během sestupné hrany hodin **CLK** logika odhalí logickou 1 (high) na signálu **/SEL** a logickou 0 (low) na signálu **/OE**.
7. V případě, že hodnota signálu **/OE** jde do logické 1 (high), datové linky mají změnu k vysokému impedančnímu stavu.
8. V tomto případě nastává druhá resetovací podmínka pro bránicí logiku. Jestliže logika objeví v průběhu sestupné hrany hodin **CLK** logickou 1 na pozici signálu **/OE**, je nastaven bránicí **IHB** signál na hodnotu logické nuly.

Časové logické rozhraní a zpracování šestnácti bitových dat zajišťuje jednočipový mikropočítač, ke kterému je obvod HCTL2016 připojen dle obrázku (Obr. 2.5).



Obr. 2.5. Elektronické zapojení obvodu HCTL 2016

2.1.2.2 Jednočipový mikropočítač

2.1.2.2.1 Požadavky na mikropočítač

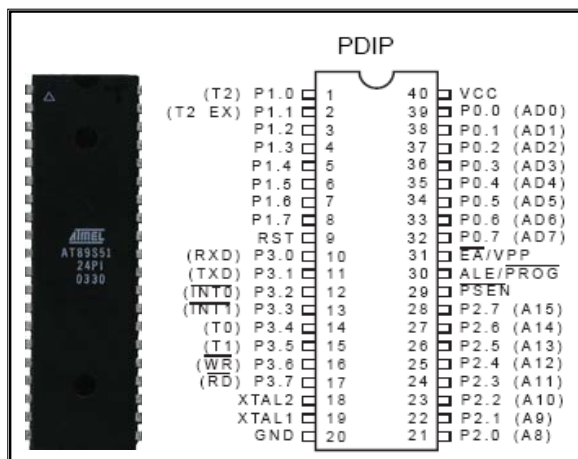
Dostatečná rychlost (tzn. velikost použitelnosti frekvence krystalu), možnost probuzení z režimu power down vnějším přerušením (standardně pouze hardwarový reset), ISP (In System Programmable) programování, dostatečná velikost paměti pro program, dostatečný počet vstupních a výstupních pinů.

2.1.2.2.2 Výběr a popis jednočipového mikropočítače – AT89S51

Pro diplomovou práci je zvolen mikroprocesor od firmy ATMEL AT89S51, který vyhovuje všem předchozím požadavkům z kapitoly 2.1.2.2.1.

Obvod AT89S51 je nízko-výkonový, CMOS 8bitový mikroprocesor s 4K bajtovou ISP (In-System Programmable) flash pamětí. Paměť mikročipu může být programována pomocí ISP (až 1000 cyklů zápisu/mazání), nebo vhodným programátorem.

Obvod poskytuje následující standardní schopnosti:



Obr. 2.6. Zapojení vývodů AT89S51

- 128bajtová vnitřní paměť RAM
- 32 programovatelných vstupů-výstupů
- duální datové pointry
- dva 16bitové časovače-čítače
- šest přerušovacích zdrojů
- plné duplexní sériové rozhraní počítače
- plné statické operace: 0 Hz–33 MHz

Tab. 2.2. Všeobecné parametry procesoru AT89S51

Parametr:	Hodnota	Jednotky
Napájecí napětí	4-5,5	[V]
Paměť	8	[kB]
Typ paměti	Flash - 4	[Kbyte]
Odběr proudu - active mod (12 MHz)	20	[mA]
Odběr proudu - idle mod (12 MHz)	6,5	[mA]
Odběr proudu - power down mod (12 MHz)	50	[μA]
Pouzdro	PDIP	

AT89S51 má k dispozici 4 obousměrné (vstupně-výstupní) porty, z nichž každý má po 8 pinů. Tyto porty jsou označovány jako P0, P1, P2, P3. Porty hardwarově sestávají ze vstupních vyrovnávacích pamětí, klopných obvodů typu D a výstupních budičů. Maximální zatížení 1 pinu portu – 10 mA. Maximální zatížení portu: P0 – 26 mA, P1, P2, P3 – 15 mA.

Tab. 2.3. Popis základních vývodů procesoru AT89S51

Pin	Označení	Popis
1 - 8	P1.0 - P1.7	Port 1
9	RST	Vstup pro nastavení výchozího stavu mikropočítače
10 - 17	P3.0 - P3.7	Port 3
16	WR	Řídící signál pro zápis do externí paměti
17	RD	Řídící signál pro čtení z externí paměti
18, 19	XTAL1, XTAL2	Připojení oscilátoru
20	GND	Zem
21 - 28	P2.0 - P2.7	Port 2
29	PSEN	Řídící výstup pro čtení z externí paměti programu
30	ALE	Výstup řídicí zápis nižší části adresy do obvodu LATCH
39 - 32	P0.0 - P0.7	Port 0
40	Vcc	Napájecí napětí

Tab. 2.4. Popis portů procesoru AT89S51

Port 0 (P0.0 - P0.7)	Jako jediný nemá ve výstupním budiči zvyšovací odpor - obvod s otevřeným kolektorem. Pokud jej chceme použít jako obecné vstupy / výstupy, musíme k pinům připojit externí odpory, jejichž druhý konec připojíme na +Ucc. Často se využívá při styku s vnější pamětí.
Port 1 (P1.0 - P1.7)	Nemá žádnou alternativní funkci, díky tomu jej můžeme bez nějakého omezení libovolně používat.
Port 2 (P2.0 - P2.7)	Kromě své normální funkce může využívat spolu s portem 0 ke komunikaci s externí pamětí
Port 3 (P3.0 - P3.7)	Všechny piny portu 3 jsou vícefunkční. Kromě standardní funkce navíc tento port poskytuje své piny pro potřeby jiným vnitřním obvodům mikroprocesoru. Seznam pinů a jim příslušné alternativní funkce zobrazuje následující tabulka (Tab. 2.5).

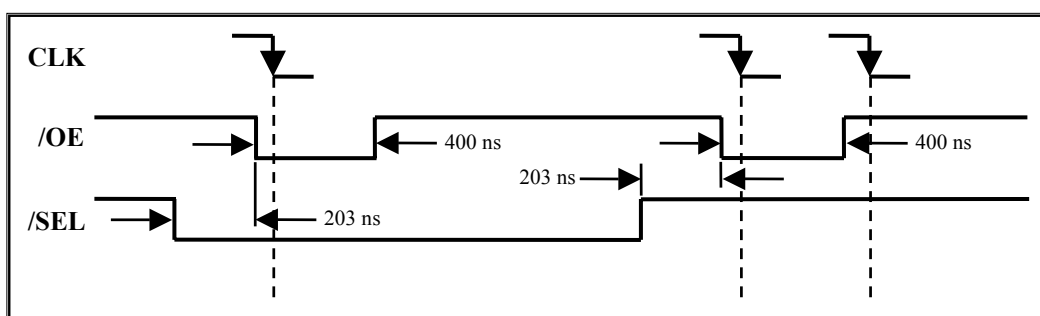
Tab. 2.5. Rozšířený popis portu 3

Pin	Označení	Alternativní funkce
P3.0	RXD	Sériový vstup
P3.1	TXD	Sériový výstup
P3.2	INT0	(Externí přerušení)
P3.3	INT1	(Externí přerušení)
P3.4	TO	Časovač / Čítač 0 externí vstup
P3.5	T1	Časovač / Čítač 1 externí vstup
P3.6	WR	Externí pulsový signál - zápis dat
P3.7	RD	Externí pulsový signál - čtení dat

2.1.2.2.3 Výběr krystalu pro mikropočítač AT89S51

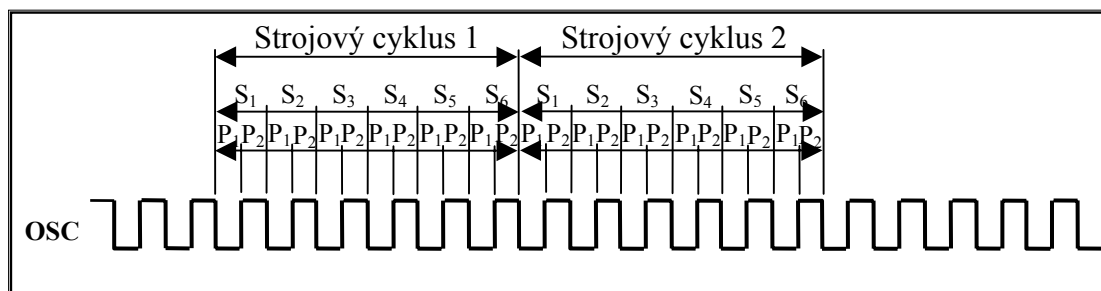
Mikroprocesor je taktován kmitočtem z externího oscilátoru. V úvahu pro výběr vhodné frekvence oscilátorového krystalu je důležité vzít dodržení časových podmínek obvodu HCTL 2016.

Z obrázku (Obr. 2.7) je vidět, že nejdelší časový úsek trvá 400 ns. A to v okamžiku, dochází-li k přepínání řídicího signálu /OE. Aby byla dodržena tato časová podmínka, je nutné zvolit vhodnou frekvenci krystalu. Většina instrukcí mikropočítačů 8051 se provádí v jednom strojovém cyklu.



Obr. 2.7. Načasování čtecího cyklu obvodu HCTL 2016

Strojový cyklus má 6 stavů označených S_1 až S_6 . Každý stav se dělí na dvě poloviny, první polovina označená jako fáze P_1 , druhá P_2 . Celkem tedy 12 period oscilátoru dle obrázku (Obr. 2.8).



Obr. 2.8. Časování procesoru

Výpočet délky jednoho strojového cyklu je stanoven pomocí frekvence oscilátoru, která je zvolena $f_{CPU} = 24 \text{ MHz}$.

$$\text{Délka strojového cyklu} = 12 \cdot T, \quad (2.1)$$

kde T je perioda oscilátoru:

$$T = \frac{1}{f_{CPU}} = \frac{1}{24 \text{ MHz}} = \underline{41,6 \text{ ns}}. \quad (2.2)$$

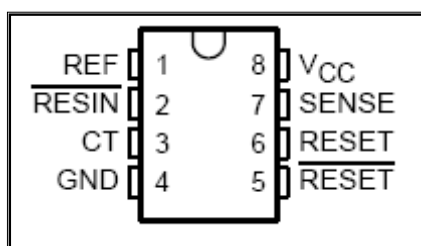
Pak tedy

$$\text{Délka strojového cyklu} = 12 \cdot 41,6 \cdot 10^{-9} = 500 \text{ ns}.$$

Tím, že délka jednoho strojového cyklu (500 ns) je větší než časový úsek pro přepínání signálu /OE (400 ns), je vhodné použít krystal se zvolenou frekvencí 24 MHz.

Procesor je schopen samostatné činnosti po připojení krystalu na vývody XTAL1 a XTAL2, napájecího napětí 5 V na vývod Ucc a připojení resetovacího (nulovacího) obvodu na vývod RST.

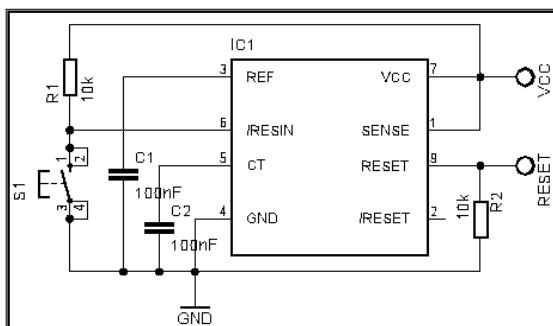
2.1.2.2.4 Resetovací obvod - TL7705



Resetovací příslušenství je tvořeno obvodem TL7705, který je vybaven přímým a negovaným výstupem (RESET, /RESET). Ten je po připojení napájení aktivní (RESET = 1, /RESET = 0) tak dlouho, dokud napájecí napětí nedosáhne hodnoty 4,55 V (prahové napětí). Oba

Obr. 2.9. Obvod TL7705

výstupy je třeba opatřit zdvihacím (/RESET), nebo snižovacím (RESET) rezistorem, protože se jedná o výstupy s otevřeným kolektorem, nebo emitorem. Napájecí napětí pro mikrořadič je snímáno vstupem SENSE. REF je výstup zabudované napěťové reference



(2,53 V), pro správnou funkci musí být toto napětí blokováno kondenzátorem s kapacitou 100 nF. Vstup /RESIN umožňuje ručně vygenerovat nulovací impuls. Tento vstup je aktivní v logické 0. CT je vstup pro časovací kondenzátor.

Obr. 2.10. Zapojení resetovacího obvodu TL7705

2.1.2.2.5 Podpůrné obvody mikroprocesoru

Jelikož procesor AT89S51 je pro programování dostatečně vybaven velkou pamětí, je proto EPROM 24C02 jen nadstandardem pro další vývoj, či rozšíření programovacích prvků modelu. Obvod je připojen k procesoru sériovou sběrnici I²C.

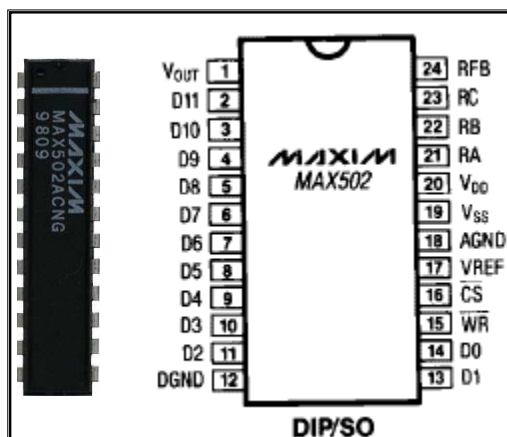
Pro případnou sériovou komunikaci mikropočítače s PC je do obvodu také zařazen napěťový převodník úrovní RS232 / TTL. Ten obsahuje dvě dvojice oddělovačů konvertujících napěťové úrovně. Tento obvod disponuje DC-DC měničem na principu nábojové pumpy a pomocí externích kondenzátorů si vyrábí napětí +10 V, -10 V.

2.1.2.3 D/A převodníky

Činností D/A převodníků je převod digitálních signálů vyhodnocených mikročipem na analogový údaj. Výstupem je analogová hodnota polohy a rychlosti servopohonu.

2.1.2.3.1 D/A převodník (Digitálně/Analogový) - MAX 502

Integrovaný obvod MAX502 je 12bitový, vícenásobný digitálně-analogový měnič



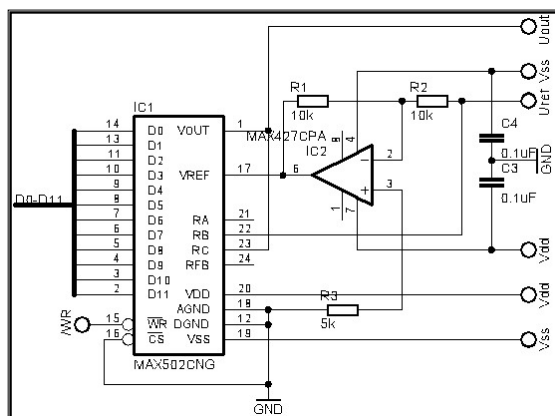
s výstupním zesilovačem. D/A převodník má lehce spojitelné dvanácti bitové rozhraní s mikroprocesorem. Všechna vstupní logika je srovnatelná s klopnými obvody kompatibilní s TTL a +5 V CMOS logickou úrovní. Vnitřní kompenzace a malé vstupní offsetové elektrické napětí výstupního zesilovače poskytne výstupní napětí v rozsahu ± 10 V, při proudu až 5 mA.

Obr. 2.11. D/A převodník MAX 502

Tab. 2.6. Popis vývodů D/A převodníku MAX502

Pin	Označení	Funkce
1	V_{out}	Výstupní napětí
2-11	D11-D2	Datové bity 2-11 (MSB)
12	DGND	Digitální zem
13, 14	D1, D0	Datové bity 0-1 (LSB)
15	/WR	Write, vstup aktivní v log. 0
16	/CS	Chip-Select, vstup aktivní v log. 0
17	VREF	Vstupní referenční napětí
18	AGND	Analogová zem
19	V_{ss}	Záporné napájecí napětí
20	V_{dd}	Kladné napájecí napětí
21	RA	Rezistor, RA = 4RFB
22	RB	Rezistor, RB = 2RFB
23	RC	Rezistor, RC = 2RFB
24	RFB	Zpětnovazební rezistor

Pro bipolární binární operace je s obvodem zapojen operační zesilovač MAX 427. Tato aplikace vytvoří z referenčního napětí U_{ref} invertující vstup pro D/A převodník



(VREF) viz obrázek (Obr. 2.12). Zisková chyba může být nastavena procentuelní změnou rezistorů R1 a R2. Vstupní registry obvodu MAX502 jsou řízené standardními signály CHIP-SELECT (/CS) „výběr čipu“ a WRITE (/WR) „zápis“. Pro samostatnou operaci jsou /CS a /WR vstupy uzemněny (log. 0), dále pak viz pravdivostní tabulka (Tab. 2.7).

Obr. 2.12. Zapojení D/A převodníku MAX502

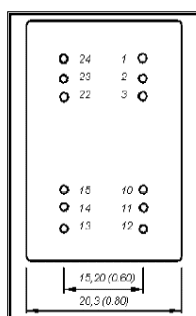
Tab. 2.7. Pravdivostní tabulka signálů /WR, /CS

/WR	/CS	Funkce
H	X	Žádná funkce
X	H	Žádná funkce
L	L	Vstupní registr je otevřený
L	R	Vstupní registr je blokován
R	L	Vstupní registr je blokován

H...(high) log.1, L...(low) log.0, R...náběžná hrana, X...beze stavu

Vzhledem k tomu, že D/A převodníky MAX 502 a operační zesilovač MAX 427 potřebují být napájené jak kladným elektrickým napájecím napětím U_{dd} , tak záporným U_{ss} , je nutné napětí ověřit a upravit výstupní napájecí napětí z PC analogové karty $U_{cc} = 5\text{ V}$ na symetrické napětí $\pm 12\text{ V}$.

2.1.2.4 DC/DC měnič TEM3 - 521



Pro aplikaci zesílení a invertování napájecího napětí je vybrán elektronický obvod TEM3 - 521. Tento DC/DC měnič vytvoří symetrické elektrické napětí, které využijeme pro napájení D/A převodníků a operačních zesilovačů (U_{dd} , U_{ss}). Elektrické parametry vstupních a výstupních veličin obvodu udává tabulka (Tab. 2.8).

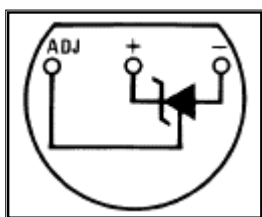
Obr. 2.13. Vývody DC/DC

Tab. 2.8. Elektrické parametry DC/DC měniče

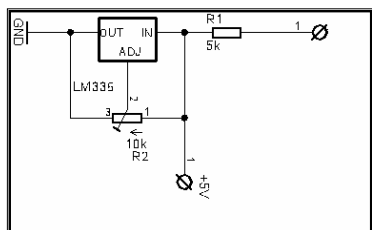
Parametr	Hodnota	Jednotky
Vstupní napětí	5	[V]
Výstupní napětí	± 12	[V]
Výkon	3	[W]
Výstupní proud	125	[mA]

2.1.2.5 Zdroj referenčního napětí LM336

Pro správnou a přesnou funkčnost D/A převodníků je nutné přesného referenčního napětí. Vybraný integrovaný obvod má nastavitelnost referenčního napětí od +4 V do +8 V. Vyznačuje se nízkým teplotním součinitelem, dynamickou impedancí 0,6 Ω a šířkou operačního proudu od 600 μA do 10 mA. Obvod je integrován do pouzdra TO-92.



Obr. 2.14. Vývody LM336



Zapojení uvedené na obrázku (Obr. 2.15) znázorňuje aplikaci nastavení žádaného referenčního napětí $U_{\text{Ref}} = +8 \text{ V}$ odporovým trimrem R2 (10 K Ω).

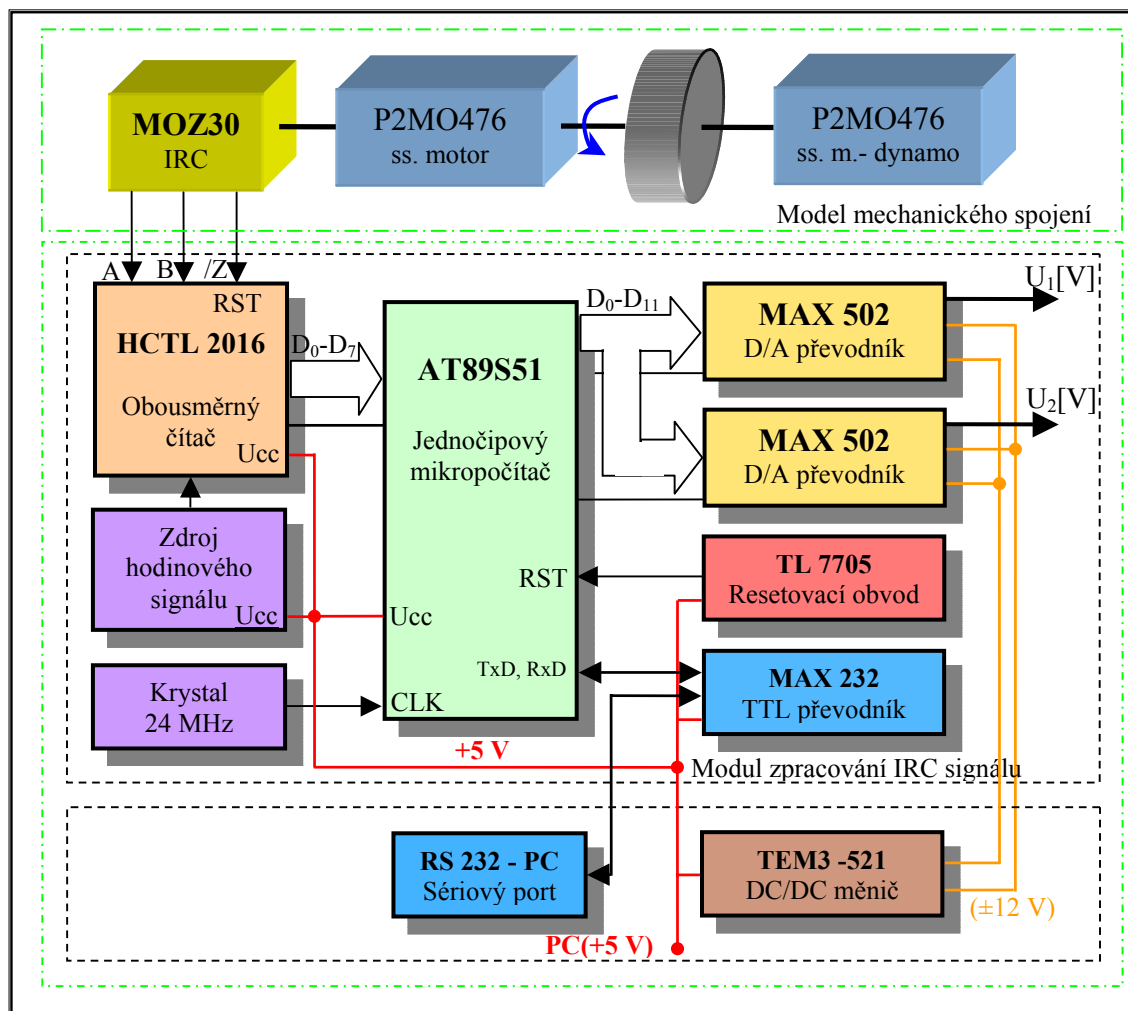
Obr. 2.15. Elektrické zapojení obvodu LM336

2.1.3 Závěr návrhu snímání a vyhodnocení signálů z IRC snímače

Výběrem vhodných elektronických součástek dle předchozích kapitol, je navrhována a sestavena snímací jednotka IRC signálů viz obrázek (Obr. 2.16). Elektronické schéma je uvedeno v příloze A 3.1.

Signály *A*, *B* a *Z* z inkrementálního čidla snímá obousměrný dekodér/čítač HCTL 2016. Dle kapitoly (2.1.2.1) se tyto signály filtrují, dekódují a následně čítají. Výstupní 16bitové slovo zpracovává jednočipový mikropočítač AT89S51, který je připojen portem P0 k osmibitovému rozhraní obvodu HCTL 2016. Procesor se také dále stará o časovací logiku ovládacím signálem vysílaného na pin HCTL 2016 (*SEL*) viz kapitola (2.1.2.3). 16bitové slovo je rozděleno do dvou bajtů po 8bitech. Vyšší bajt nazýván jako HIGH

BYTE (8-15 bitů), nižší LOW BYTE (0-7 bitů) dle obrázku (Obr. 2.17). Zpracované data na výstupu mikroprocesoru jsou D/A převodníky převedena na napětí úměrné poloze a napětí úměrné rychlosti otáčení hřídele ss. motoru.



Obr. 2.16. Blokové schéma zpracování IRC signálu



Obr. 2.17. Rozdělení 16bitového datového slova

2.1.4 Softwarová realizace mikropočítače

V této části práce je popsána struktura programu v assembleru, která se zabývá o vlastní snímání a vyhodnocení polohy, rychlosti otáček servopohonu.

2.1.4.1 Nastavení časovače

Na začátku snímacího cyklu je nejprve nastaven časovač mikroprocesoru AT89S51 tak, aby každá měřicí smyčka trvala 400 μ s. Tzn., že časovač je nastaven na čítání do 400 μ s, mezitím je provedeno hlavní tělo programu a na konci se čeká, až časovač doběhne do určené doby (dokud časovač nenastaví flag přetečení, tzn. nastavení bitu TF0 řídicího registru TCON časovače). 400 μ s je čas, který je zaručeně větší, než-li nejdelší doba celého těla programu pro snímání polohy a rychlosti.

Výpis části kódu pro nastavení časovače:

```
mov tl0, #0efh
mov th0, #0fch
mov tmod, #00000001b ; nastavení registru TMOD, režim časovač/čítač->16bit. časovač
setb TR0 ; nastavení pozice TCON.4 registru-řídící bit běhu časovače-spuštění
```

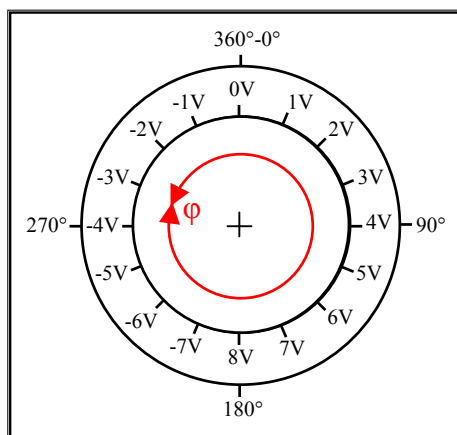
2.1.4.2 Čtení dat z převodníku

V další části kódu je popsáno čtení dat z převodníku HCTL2016, který je připojen k mikroprocesoru portem P0:

Výpis části kódu pro čtení dat z dekodéru:

```
anl IRCC, #AndMask ; provedení vypnutí /OE a SEL signálu
;; přečtení hodnoty z portu P0 - vyšší bajt
mov a, IRCD ; uložení IRC Dat do akumulátoru, (IRCD = port P0)
mov irch, a ; uložení akumulátoru do proměnné irch = horní bajt načtené hodnoty z IRC
;; přečtení hodnoty z portu P0 - nižší bajt
setb IRCC.IRCSel ; nastavení SEL signálu do log. 1
mov a, IRCD ; uložení IRCD do akumulátoru
mov ircl, a ; uložení akumulátoru do proměnné ircl = dolní bajt načtené hodnoty z IRC
setb IRCC.IRCOE ; nastavení /OE signálu do log. 1
```

2.1.4.3 Stanovení úhlové polohy hřídele stejnosměrného motoru



Mikroprocesor provede přenos dat z IRC dekodéru na vstupní dvanácti bitové piny D/A převodníku. Tyto binární data jsou převedena na analogové napětí úměrné poloze. Údaj o poloze je vyhodnocen elektrickým napětím, v napětovém rozsahu od +8 V do -8 V. Tento rozsah znázorňuje otočení hřídele motoru o 360 °, tedy o jednu otáčku viz obrázek (Obr. 2.18).

Obr. 2.18. Napětový údaj o poloze

Jeden stupeň natočení hřídele motoru je roven přibližně 0.0444 V. Naopak údaj jednoho voltu udává polohu natočení o cca 22,5 °.

Výpis části kódu zajišťující snímání polohy:

```

mov a, ircl          ; uložení IRC hodnoty dolního bajtu do akumulátoru
mov posl, a          ; z akumulátoru uložení do proměnné posl (Pozice Low)
mov a, irch          ; uložení IRC hodnoty nižšího bajtu do akumulátoru
mov posh, a          ; z akumulátoru uložení do proměnné posh (Pozice High)
;; nastává datový přenos
; na port P2 = dadal (DA Data Low) - nižší bajt
mov a, posl
mov dadal, a          ; datový přenos pozice low na port P2
; na port P1 = dadah (DA Data High) - vyšší bajt
mov a, posh
swap a                ; nibble z nižší pozice jsou prohozena na vyšší nibble
mov dadah, a          ; datový přenos pozice high na port P1
; zapsání na první D/A převodník (wr1 = port P3.4)
clr wr1               ; nulování signálu WRITE A/D převodníku
nop                   ; časové zpoždění jednoho cyklu (No Operation)
nop
setb wr1              ; nastavení signálu WRITE na log. 1

```

2.1.4.4 Snímání rychlosti stejnosměrného motoru

2.1.4.4.1 Nadefinování rychlostních konstant v assembleru

Na začátku programu jsou definovány rychlostní konstanty:

- Max_Ot** **equ 120d** ... Maximální otáčky nastaveny na 120 ot/s = 7200 ot/min.
- Max_Imp** **equ 4096d** ... Maximální rozlišení IRC dekodéru, 1024 pulzů×4 = 4096 pulzů/otáčku.
- Del_Cy** **equ 400d** ... Délka cyklu v mikrosekundách - doba během které se počítá, kolik přišlo pulzů z IRC dekodéru.
- Max_DA** **equ 2048d** ... Rozlišení DA převodníku.
- Max_Cy** Maximální počet průchodů cyklem, abychom měřili ještě nějakou (nenulovou) rychlost.

Jelikož assembler umí jen dvoubajtové výpočty, je proveden ruční výpočet maximálního počtu průchodu cyklem Max_Cy. Tím, že jsou maximální otáčky nastaveny na 120 ot/s a maximální rozlišení IRC dekodéru je 4096 pulzů/ot, je počet pulzů za jednu sekundu roven:

$$P_1s = Max_Ot \cdot Max_Imp = 120ot/s \cdot 4096p/ot = \underline{491520p/s}. \quad (2.3)$$

Počet pulzů pro délku jednoho cyklu:

$$P_cy = P_1s \cdot Del_Cy = 491520p/s \cdot 400 \cdot 10^{-6}s = \underline{196pulzů}. \quad (2.4)$$

Maximální počet průchodu cyklem při rozlišení D/A převodníku:

$$Max_Cy = \frac{Max_DA}{P_cy} = \frac{2048}{196} = \underline{10,4}. \quad (2.5)$$

2.1.4.4.2 Stanovení rychlosti

V první části podprogramu o snímání rychlosti je nejprve odečtena od nové snímané hodnoty z IRC dekodéru stará pozice. Jelikož assembler pracuje v dvoubajtových počtech, musí se počítat s carry bitem.

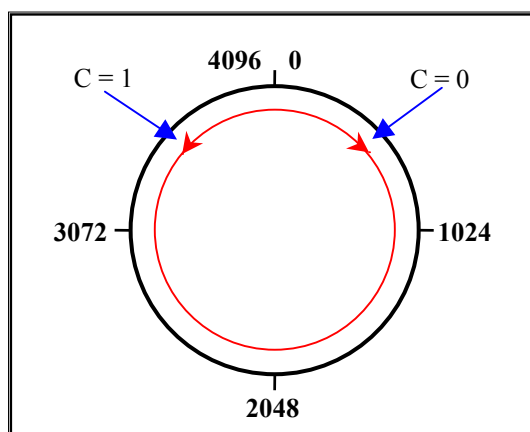
Výpis části kódu zajišťující odečet nové pozice od staré

```
mov a, ircl           ; nová pozice je uložena do akumulátoru (ircl = dolní bajt načtené hodnoty
                      ; z IRC)
subb a, ircol         ; odečítání staré pozice od nové
mov speedl, a         ; výsledek je uložen do proměnné speedl
```

mov a, irch ; totéž je provedeno s vyšším bajtem irch
 subb a, ircoh ; odečítání staré pozice od nové
 mov speedh, a ; výsledek je uložen do proměnné speedh
 mov a, irch ; pro další výpočty je uložen horní bajt minulé polohy do proměnné ircoh
 mov ircoh, a
 mov a, ircl ; pro další výpočty je uložen dolní bajt minulé polohy do proměnné ircol
 mov ircol, a

2.1.4.4.3 Směr otáčení

Pro kladný a záporný směr otáčení servopohonu je nutné provést znaménkové rozhodování dat. Z předchozího odečítání pozic je údaj o směru uložen v Carry bitu. Při použití instrukce **SUBB** je *C* nastaven, když výsledek odečítání pozic podteče a nevejde se do akumulátoru. V opačném případě je *C* nulován. Je-li tedy Carry bit nulový, směr otáčení je kladný a naopak je-li nenulový je záporný.



Obr. 2.19. Nastavení Carry bitu při kladném a záporném směru otáčení

Podmínka s Carry bitem však neplatí pro změnu pozic při průchodu nulou. Tzn., že nastávají dvě varianty. Motor se točí v kladném směru, změna hodnoty IRC dekodéru je $4096 \gg 0$ a Carry bit má nenulovou hodnotu. Druhou variantou je, když směr má záporný smysl otáčení. Změna pozic IRC dekodéru je tedy $0 \gg 4096$ a Carry bit je nulový.

Je-li Carry bit a horní bajt výsledku rozdílu pozic nulový, směr otáčení servopohonu je kladný. Naopak, je-li Carry bit nulový a horní bajt IRC pozic je nenulový dochází k přetočení přes nulu v kladném směru otáčení servopohonu. Pak je od výsledné hodnoty rozdílu pozic odečtena hodnota 4096 a nastaven opačný směr.

Je-li Carry bit nenulový a horní bajt výsledku rozdílu pozic nulový, směr otáčení servopohonu je záporný. K přetočení přes nulu v záporném směru otáčení servopohonu

dochází, je-li Carry bit a horní bajt IRC dat dekodéru nenulový. K výsledné hodnotě rozdílu pozic musí být přičtena hodnota 4096 a nastaven opačný směr.

Výpis části kódu zajišťující znaménkové rozhodování:

```
Jc dirC          ; skok na návěští (dirC) při nenulovém příznaku přenosu
clr dir          ; nulování Carry bitu
mov a, speedh     ; načtení horního bajtu rychlosti do akumulátoru
jz dirCE          ; horní bajt je nulový, normální kladná hodnota, skok na návěští dirCE
; horní bajt je nenulový, prošlo se nulou
call Sub4096      ; volání podprogramu Sub4096 (od výsledné hodnoty rychlosti je
odečtena pozice 4096)
call Abs          ; volání podprogramu Abs (absolutní hodnota)
setb dir          ; nastaven opačný směr
jmp dirCE         ; skok na návěští dirCE (výpočet rychlosti)
dirC:             ; je carry - točí se asi záporně
setb dir ;
call Abs          ; absolutní hodnota
; horní bajt je nenulový, prošlo se nulou
mov a, speedh     ; načtení horního bajtu rychlosti do akumulátoru
jz dirCE ; horní bajt je nulový, normální kladná hodnota, skoč na návěští dirCE (výpočet rychlosti)
; horní bajt je nenulový, prošlo se nulou
call SubFrom4096  ; volání podprogramu SubFrom4096 (k výsledné hodnotě rychlosti je
přičtena pozice 4096)
clr dir          ; nastaven opačný směr
```

2.1.4.4 Měření rychlosti

Měření rychlosti pracuje v cyklech trvajících 400 μ s, kde jsou počítány pulzy z IRC dekodéru. Přejde-li během jednoho průchodu (400 μ s) víc pulzů, rychlost se vypočte pouhým dělením dané pozice a času. Pokud se ale IRC snímač otáčí pomalu tzn., že přijde jeden pulz teprve až za několik cyklů, musejí se slepě sčítat jak pulzy (speed), tak časové průchody (counter) cyklem. Napočítá-li se aspoň jeden pulz dojde k podělení a tím je získána rychlost. Pokud během koeficientu Max_Cy nepřijde žádný pulz, rychlost je nulová. Ve chvíli, kdy po dlouhém stání přijde pulz, se počet cyklů bere jako Max_C. Jinak řečeno, cykly se počítají od jedné do Max_Cy, pak už není countter zvětšován.

Výpis části kódu zajišťující výpočet rychlosti:

dirCE:

```
inc counter          ; na slepo zvětším čítač cyklů
mov a, speedl        ; pokud je změna IRC dat, musí se projevit v proměnné speedl
jnz nestoji          ; pokud ne, skok na sekci stojí
mov a, counter
cjne a, #(Max_Cy+1), cykli ; pokud není dosažena kritická hranice, zvětšen čítač
dec counter          ; pokud je dosažena kritická hranice, tak je zase zmenšen
cykli:
```

nestoji:

```
mov b, #Max_Cy        ; uložení koeficientu Max_Cy do pomocného registru b
mul ab                ; vynásobení počet pulzů koeficientem Max_Cy
mov r1, b              ; vyšší bajt do r1
mov r0, a
mov r2, counter        ; dělení počtem nastrádaných cyklů
mov r3, #0
call UDIV16            ; volání podprogramu UDIV16 (dělení dvou
                        ; šestnáctibitových čísel bez znaménka)
mov speedh, r1          ; uložení vyššího bajtu rychlosti do speedh
mov speedl, r0          ; uložení nižšího bajtu rychlosti do speedl
mov counter, #0
```

2.1.4.4.5 Znaménková korekce dat

Tím, že údaj o rychlosti je vyhodnocen analogovým napětím v rozsahu ± 8 V musí být provedena znaménková korekce dat. Tzn., je-li směr otáčení motoru kladný, je přičtena k rychlosti hodnota 2048. Naopak pro záporný směr je tato hodnota od rychlosti odečtena.

Př. Výpis části kódu pro přičtení hodnoty 2048 k rychlosti:

```
jb dir, eplus          ; je-li kladný směr přičte se 2048
mov a, speedh          ; horní bajt rychlosti uložen do akumulátoru
add a, #00001000b      ; přičtení hodnoty 2048
mov speedh, a           ; uložení akumulátoru zpět do horního bajtu rychlosti
eplus:
```

2.1.4.4.6 Zápis rychlosti na D/A převodník

Konečnou fází softwarového vyhodnocení rychlosti otáčení servopohonu je zápis dat do D/A převodníku. Obvod MAX502 je připojen k procesoru portem P1 a P2.

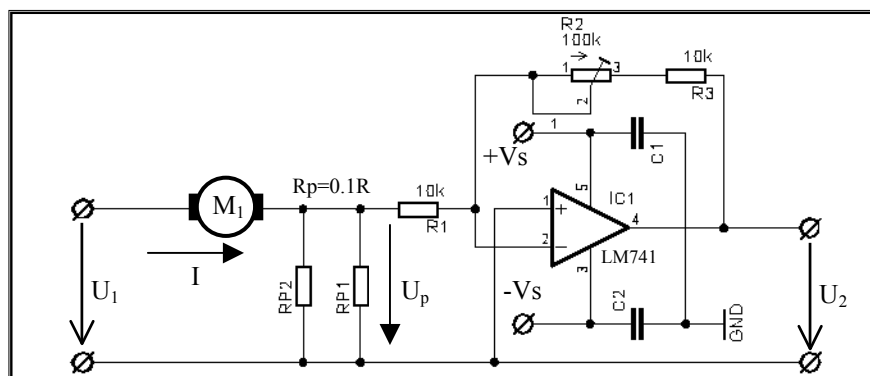
Výpis části kódu pro zápis na druhý D/A převodník:

```

mov a, speedl      : načtení horního bajtu rychlosti do akumulátoru
mov dadal, a       ; načtení akumulátoru na port P2
mov a, speedh      ; načtení dolního bajtu rychlosti do akumulátoru
swap a            ; nibble z nižší pozice jsou prohozena na vyšší nibble
mov dadah, a       ; načtení akumulátoru na port P1
clr wr2           ; nulování signálu WRITE A/D převodníku
nop               ; časové zpoždění jednoho cyklu (No Operation)
nop
setb wr2          ; nastavení signálu WRITE na log. 1
  
```

2.2 Snímání proudu v kotvě ss. motoru

Jak už bylo řečeno v úvodu diplomové práce, v regulační kaskádě stejnosměrného motoru se pracuje na nejnižším stupni s regulací proudu. Proto v této kapitole je navrhnut obvod, který udává informaci elektrického napětí úměrné proudu kotvou ss. motoru. Proudový snímač je vytvořen pomocí paralelního zapojení dvou malých výkonových rezistorů ($R_p = 0,22 \Omega // 0,22 \Omega$ (5 W)) a operačního zesilovače LM741 v invertujícím zapojení. Snímání proudu je v tomto případě převedeno na měření úbytku napětí na známém rezistoru R_p , viz elektronické zapojení na obrázku (Obr. 2.20).



Obr. 2.20. Snímání proudu v kotvě pomocí OZ a malého rezistoru

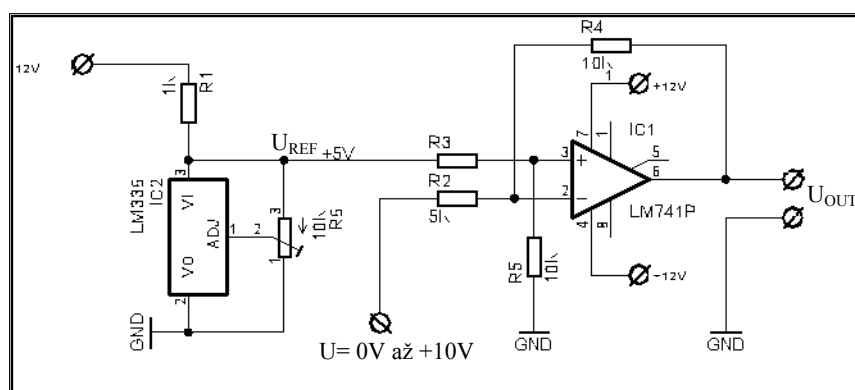
Pro výběr vhodných rezistorů v invertujícím zapojení operačního zesilovače se vychází ze základní rovnice obvodu:

$$U_2 = U_p \cdot \frac{R_3 + R_2}{R_1} = I \cdot R_p \cdot \frac{R_3 + R_2}{R_1}, \quad (2.6)$$

kde U_p je úbytek napětí na známém rezistoru a I je proud kotvy stejnosměrného motoru. Poměr rezistorů $(R_3 + R_2)/R_1$ určuje zesílení operačního zesilovače. Rezistory R_1 , R_3 jsou zvoleny 10 kΩ. Odporovým trimrem $R_2 = 100$ kΩ je možné nastavení zesílení operačního zesilovače a tím určení vhodného poměru pro výstupní napětí úměrné proudu v kotvě stejnosměrného motoru.

2.3 Převod polarity výstupního analogového signálu z PCI-1711

Nevýhodou multifunkční karty **PCI-1711** oproti PC kartě MF 604 je její analogový napěťový výstup, který je pouze v rozsahu 0÷10 V. Původní ročníkový model byl navržen pro výstupní napětí od -10 V až +10 V. Proto dalším návrhem pro reverzaci servopohonu je upravení výstupního napětí z analogové karty **PCI-1711**. Mezi výkonový operační zesilovač LM12CLK a analogový výstup je zapojen pomocný OZ (LM741). Ten v diferenčním zapojení odečítá od konstantního napětí +5 V, které je vytvořeno referenčním zdrojem LM336, výstupní napětí z karty **PCI-1711** viz obrázek (Obr. 2.21). Tímto aplikačním zapojením získáme na výstupu požadované napětí v rozsahu ±10 V.



Obr. 2.21. Zapojení předzesilovače

Napěťové zesílení A_u vyplývá dosazením vhodných rezistorů do základní rovnice pro diferenční zesilovač

$$U_{out} = A_u \cdot (U_{REF} - U). \quad (2.7)$$

Při dodržení podmínek rovnosti rezistorů $R_2 = R_3$ a $R_4 = R_5$ se napěťové zesílení A_u rovná poměru rezistorů

$$A_u = \frac{R_2}{R_4} \quad (2.8)$$

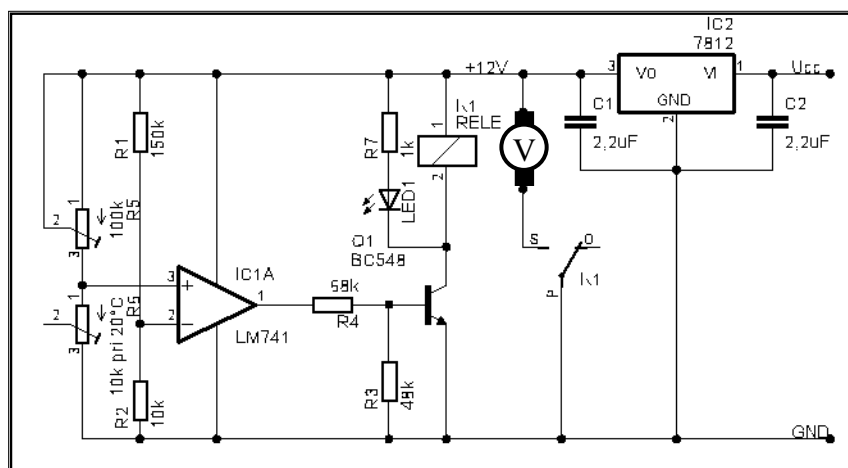
Výsledné výstupní napětí U_{out} předzesilovacího bloku je dle vztahu (2.9) v rozsahu ± 10 V.

$$U_{OUT} = \frac{R_2}{R_4} (U_{REF} - U) \quad (2.9)$$

2.4 Chlazení výkonového operačního zesilovače

Výstupní napětí z předzesilovacího bloku v rozsahu ± 10 V je zesíleno jednoduchým neinvertujícím zapojením výkonového operačního zesilovače LM12CLK. Analogový výstup je zesílen hlavně v proudové části, kde maximální proud OZ je $I_{max} = 10$ A. Napěťové zesílení $A_u = 1,2$ udává poměr rezistorů na OZ. Operační zesilovač je napájen symetrickým napětím (± 14 V) z laboratorního zdroje HC-D230-4B, který je vybaven dvěma zdroji s plynulou regulací napětí 0-30 V a s možností nastavení proudu od 0,1-4 A.

Z důvodu energie výkonové ztráty na OZ LM12CLK, která se mění v teplo, byl v ročníkovém projektu řešen problém s chlazením OZ. V závěru se ukázalo, že se operační zesilovač, který byl připevněn na chladič, stále při delším zatížení zahřívá. Proto je na chladič připevněn DC ventilátor o rozměrech 80×80×25 mm. Aby ventilátor běžel, jen když je zapotřebí a zbytečně nedělal hluk, použijeme zapojení s odporovým teplotním čidlem a operačním zesilovačem LM741. OZ je zde zapojen jako komparátor, to znamená, že je bez zpětných vazeb. Porovnává tedy dvě napětí, referenční, které je nastavené odporovým děličem R_1 , R_2 na přibližně 1 V s napětím druhého odporového děliče R_5 , $R_{čidlo}$. Jakmile napětí na vstupu $In+$ operačního zesilovače přesáhne hodnotu napětí na vstupu $In-$, to znamená, že se operační zesilovač otevře naplno a dále již nereaguje na změny napětí na vstupu, je na výstupu maximální napětí (saturační).

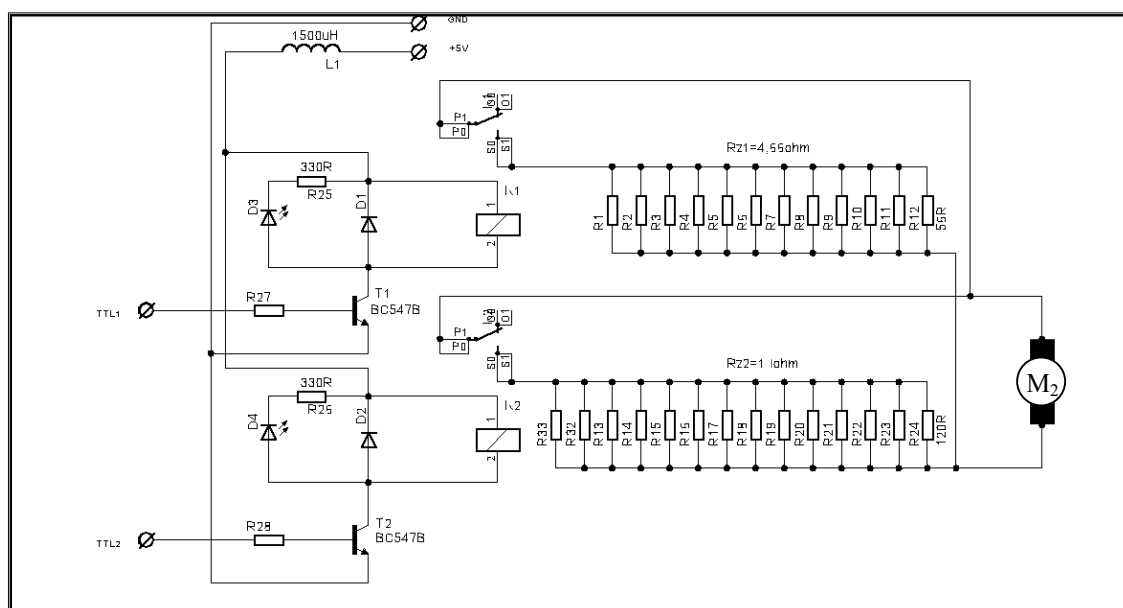


Obr. 2.22. El. schéma zapojení teplotního spínače

Rezistory R_4 , R_3 tvoří dělič napětí před NPN tranzistorovým spínačem (BC548). Tím, že se objeví na výstupu OZ saturační napětí, tranzistor se otevře a sepne relé. Po sepnutí reléového kontaktu se spustí ventilátor. Abychom dodrželi referenční hodnoty, musí být napájecí napětí stabilizováno. Stabilizaci napětí hlídá obvod L7812 s výstupním napětím +12 V. Pro signalizaci stavu ventilátoru je do obvodu zařazena LED dioda. Elektronické zapojení vlastního teplotního spínače je znázorněno na obrázku (Obr. 2.22).

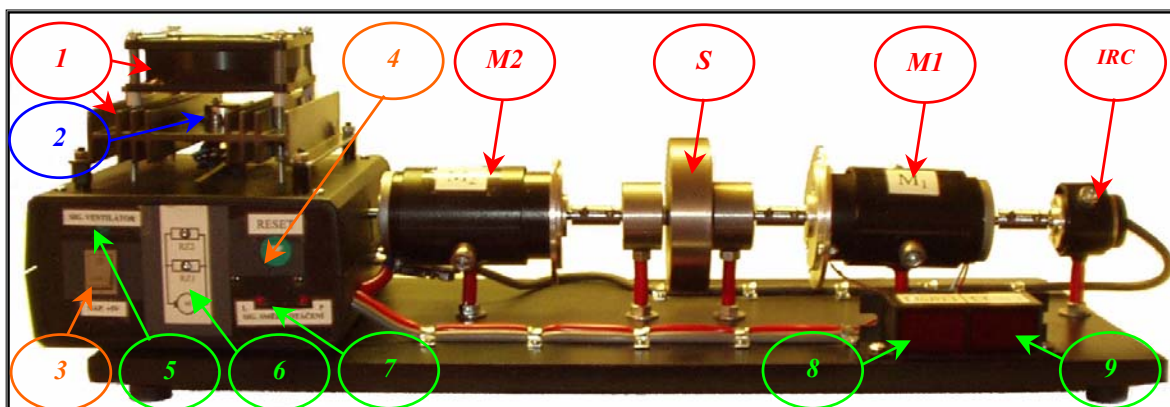
2.5 Elektrodynamická brzda

Elektrodynamická brzda je zařízení, které mění mechanickou práci na elektrický proud a tím dochází k brždění. Na svorky motoru M_2 jsou pomocí přepínacích kontaktů relé připojeny zatěžovací odpory R_{z1} (paralelní zapojení výkonových rezistorů $12 \times 120 \Omega$ (5 W), $2 \times 2,2 \Omega$ (5 W) a R_{z2} (paralelní zapojení výkonových rezistorů $12 \times 56 \Omega$ (5 W)). Oproti původní aplikaci v ročníkovém projektu je jedna z rezistorových zátěží změněna. Ke stávajícímu paralelnímu zapojení dvanácti rezistorů o hodnotě 120Ω (5 W) jsou paralelně připojeny další dva $2,2 \Omega$ (5 W) rezistory. Tím je zvětšena celková hodnota brzdného momentu motoru. Relé jsou ovládána pomocí digitálních výstupů analogové PC karty. Tyto výstupy jsou zesíleny tranzistory T1, T2 (BC547B). Sepnutí odporových zátěží je signalizováno LED diodami. Na obrázku (Obr. 2.23) je schematicky znázorněno zapojení spínané zátěže.



Obr. 2.23. Zapojení spínané zátěže pomocí TTL výstupu z analog. karty

2.6 Vizuální a mechanické zpracování servopohonu



Obr. 2.24. Mechanické zpracování servopohonu

2.6.1 Popis mechanického zapojení

M1, M2 Stejnosměrné motory **P2M0476**

S Setrvačnick

IRC Inkrementální rotační snímač **MZO30**

1 Chlazení výkonového operačního zesilovače LM12CLK

2 Operační zesilovač LM12CLK

3 Přepínač pro zapnutí a vypnutí napájecího napětí +5 V

4 Resetovací tlačítko mikroprocesoru

5 Signalizace stavu chladícího větráčku pomocí LED diody

6 Signalizace stavu rezistorových zátěží

7 Signalizace stavu směru točení servopohonu

8 3½ Digit, LED displej DMS-20PC vyhodnocující napětí na svorkách motoru M1

9 3½ Digit, LED displej DMS-20PC znázorňující proud v kotvě motoru M1

Signalizační a ovládací prvky jsou umístěny na přední straně panelu modelu. Součástí této přední desky jsou víceméně prvky udávající signalizaci, pomocí LED diod, určitého stavu servomodelu. Panel tedy obsahuje signalizaci stavu ventilátoru [5], signalizaci ohledně aktivity zapojení rezistorových zátěží [6] a signalizaci pro směr otáčení servomotoru [7]. Na panelu se také nachází přepínač [3] pro zapnutí napájecího napětí +5 V, kterým je napájena IRC snímací jednotka. Posledním ovládacím prvkem je mikrotlačítko [4], které je připojeno na resetovací vstup obvodu TL7705.

3 Popis zapojení úlohy a signálů

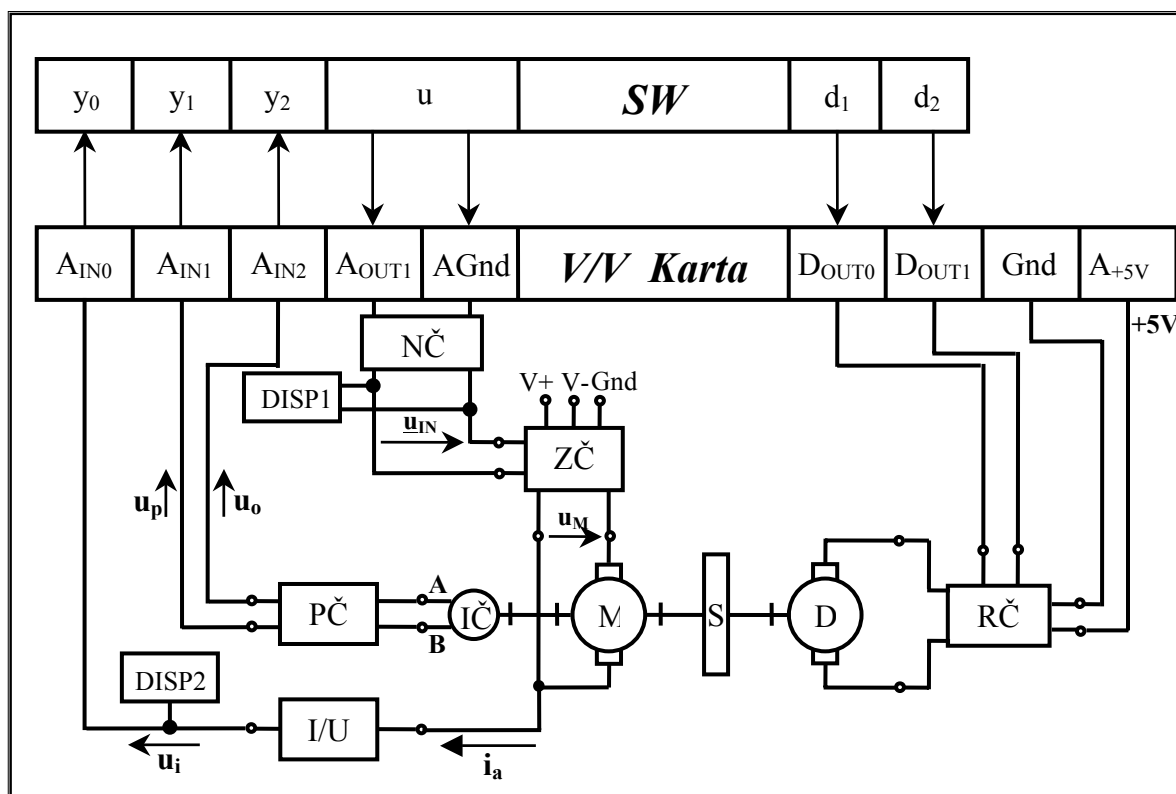
3.1 Úvod



Obr. 3.1. PC karta PCI-1711

O vzájemnou komunikaci se na straně počítače stará V/V PCI karta **PCI-1711** od firmy **Advantech**. Technologické schéma propojení je znázorněné obrázkem (Obr. 3.2). Model servopohonu lze považovat za reálný systém, který můžeme zpětnovazebně řídit. Pro výukové účely je vhodné jeho připojení

k počítači a s pomocí vytvořeného prostředí v Control Web 5 viz kapitola 4, nebo v software MATLAB s nadstavbou Simulink provádět real-time řízení, měření a zpracování signálů.



Obr. 3.2. Technologické schéma

3.1.1 Popis komponent na obrázku (Obr. 3.2)

V/V Karta - měřicí karta Advantech **PCI-1711**

$A_{IN0}-A_{IN2}$ - analogový vstup karty Advantech **PCI-1711**

- A_{Gnd} - analogová zem karty Advantech **PCI-1711**
 A_{OUT0} - analogový výstup ($0 \div 10$ V) karty Advantech **PCI-1711**
 A_{+5V} - analogový výstup (+5 V) karty Advantech **PCI-1711**
SW - používaný software (MATLAB Simulink, Control Web 5)
M - DC motor s permanentními magnety P2MO476
D - dynamo tvořené DC motorem s permanentními magnety P2MO476
IČ - inkrementální čidlo MOZ30, S - setrvačnick
ZČ - zesilovací člen (operační zesilovač LM12CLK)
RČ - elektronický rezistorový zátěžový člen
NČ - napěťový člen pro reverzaci motoru
PČ - převodní člen (signály A,B převedeny na napětí úměrné otáčkám a poloze)
I/U - člen převádějící proud kotvy na napětí
DISP1 - 3½ Digit, LED displej DMS-20PC vyhodnocující napětí na svorkách motoru M1
DISP2 - 3½ Digit, LED displej DMS-20PC vyhodnocující proud v kotvě motoru M1

3.1.2 Popis signálů soustavy

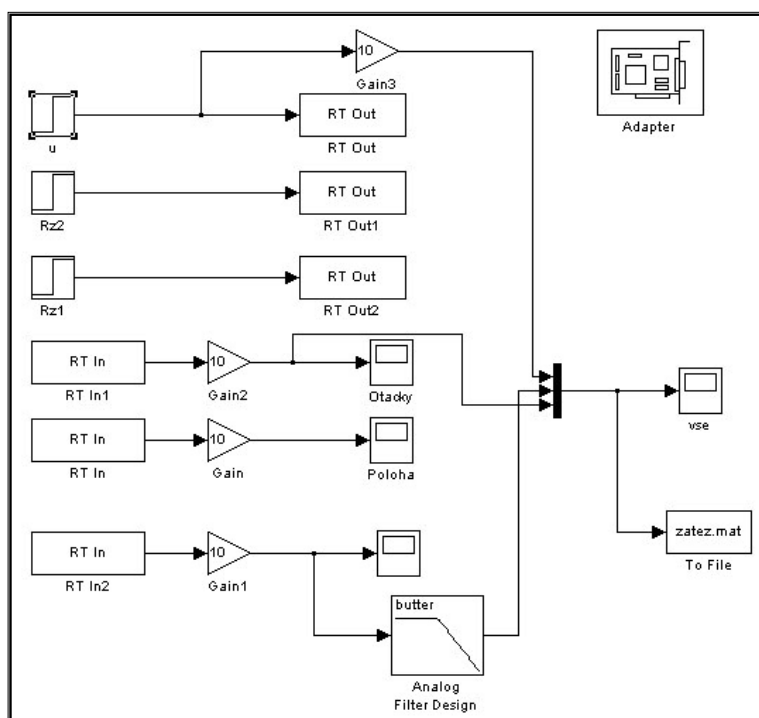
- u vstupní veličina soustavy
 u_{IN} řídicí napětí na výstupu karty, rozsah $0 \div 10$ V
 u_M napětí na svorkách motoru, rozsah $-12 \div 12$ V
 u_o napětí úměrné otáčkám servopohonu
 u_p napětí úměrné poloze natočení hřídele servopohonu
 u_i napětí úměrné proudu kotvy motoru M1
 i_a proud kotvy motoru M1
 d_1 porucha působící na soustavu, rezistorová zátěž R_{z1} spínaná digitálním výstupem
 d_2 porucha působící na soustavu, rezistorová zátěž R_{z2} spínaná digitálním výstupem
 y_0 výstupní veličina soustavy, napětí úměrné proudu v kotvě motoru
 y_1 výstupní veličina soustavy, napětí úměrné otáčkám motoru
 y_3 výstupní veličina soustavy, napětí úměrné poloze motoru
A, B signály inkrementálního snímače

3.2 Reálné měření na modelu servopohonu v prostředí MATLAB Simulink – RealTime Toolbox

Tab. 3.1. Nastavení signálů v MATLAB SIMLINKu

Veličina	I/O signál	Rozsah kanálu	Kanál	Rozsah v MATLABU	Význam
u	AO0	0 .. 10 V	1	<-1 .. 1>	Řídicí napětí motoru
d ₁	D01	{0;5} V	3	{0;1}	Spínání rezistorové zátěže Rz1
d ₂	D02	{0;5} V	4	{0;1}	Spínání rezistorové zátěže Rz2
uo	AI0	-10 V .. 10 V	1	<-1 .. 1>	Napětí úměrné otáčkám
up	AI1	-10 V .. 10 V	2	<-1 .. 1>	Napětí úměrné poloze
ui	AI2	-10 V .. 10 V	3	<-1 .. 1>	Napětí úměrné proudu v kotvě

K ovládání vstupů a výstupů karty slouží dvojice bloků **RT Out** a **RT In**. Nastavení kanálů jednotlivých bloků je provedeno dle tabulky (Tab. 3.1). Z důvodu šumu na proudové složce je do obvodu vložen dolnoproputní analogový filtr.

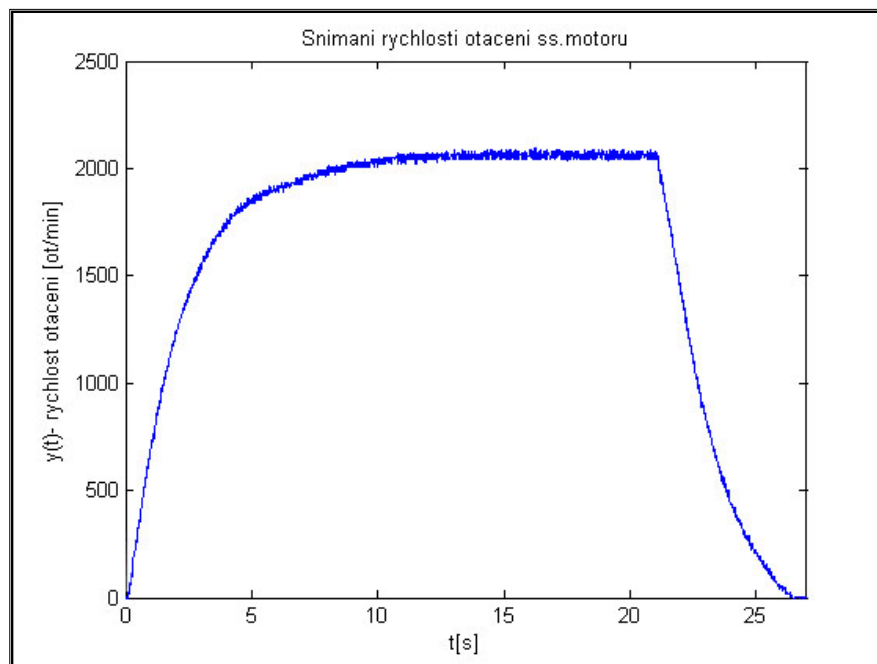


Obr. 3.3. Blokové schéma řízení modelu servopohonu

3.2.1 Odezvy výstupních signálů

Na vstup soustavy je přiveden signál u ve formě analogového napětí. Měřené výstupní analogové veličiny soustavy jsou otáčky, poloha a proud. Průběhy reálných experimentů ilustrují následující obrázky v sekundách reálného času.

3.2.1.1 Přechodová charakteristika reálného modelu

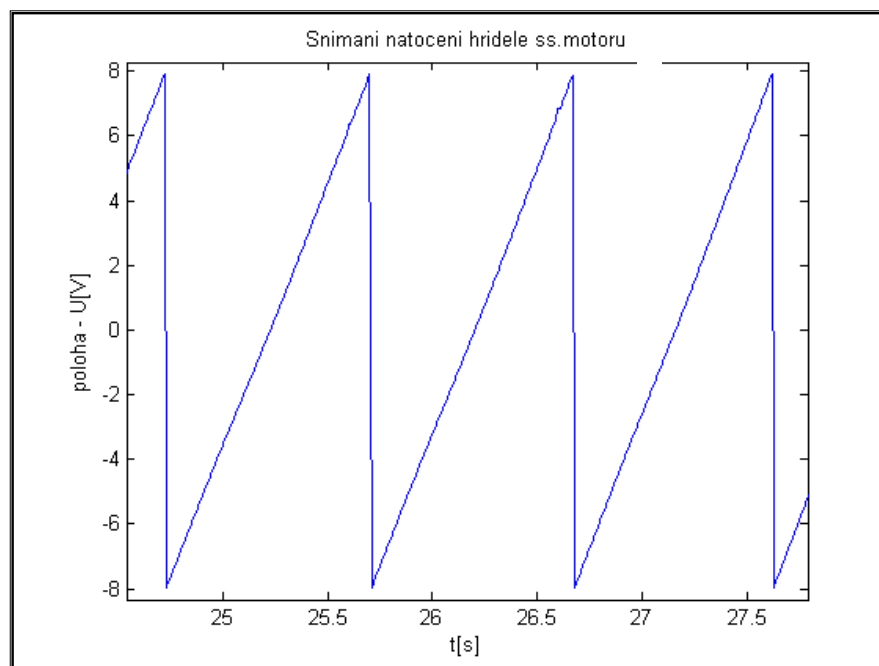


Obr. 3.4. Snímání rychlosti otáčení servopohonu

Akční veličina v podobě skokového analogového napětí u je nastavena na hodnotu 5 V. Čas měření t je 27 s. Vzorkovací perioda T_s pro snímání je nastavena na 0,01 s. Výsledné grafické znázornění rychlosti otáčení hřídele servopohonu je znázorněno na obrázku (Obr. 3.4), kde je vykreslena časová závislost napětí převedená na rychlost otáčení servopohonu.

3.2.1.2 Snímání polohy hřídele reálného modelu

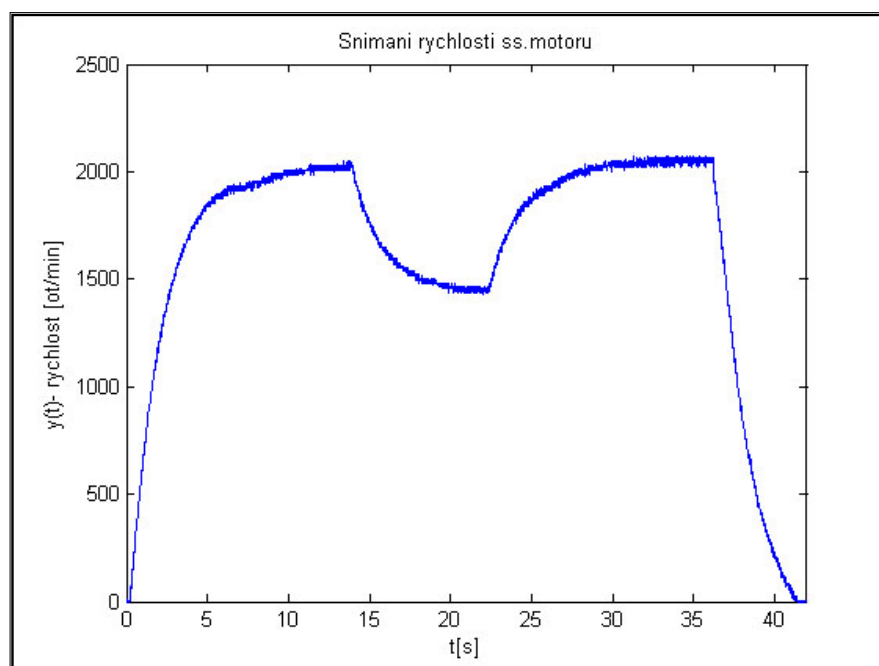
Následující obrázek (Obr. 3.5) ilustruje pilový signál snímání natočení hřídele servopohonu v časovém úseku (cca 3 s), kde údaj jedné otáčky je v napěťovém intervalu od -8 V do +8 V.



Obr. 3.5. Snímání natočení hřídele servopohonu

3.2.1.3 Vliv poruchy na rychlost otáčení hřídele modelu

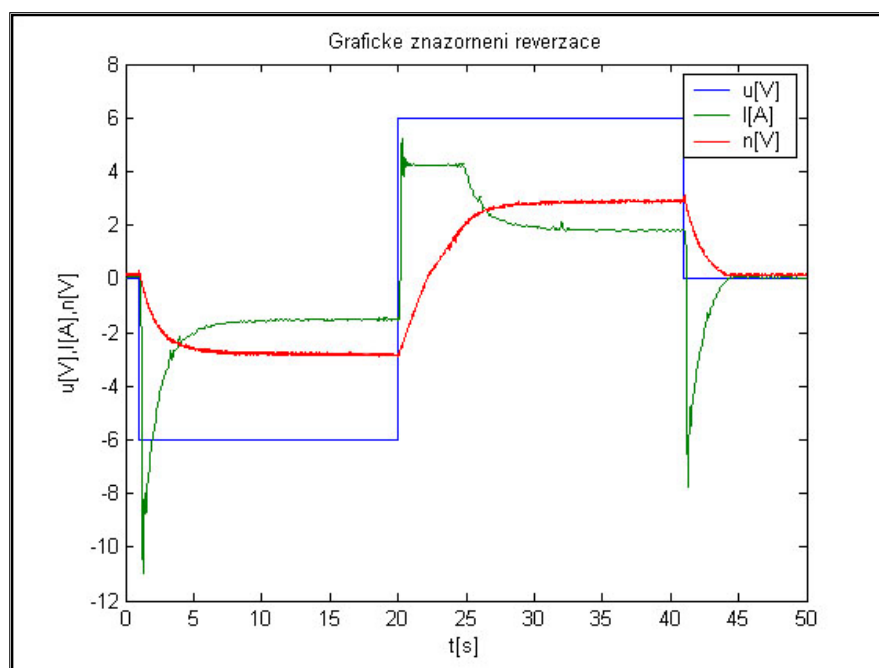
V tomto experimentu je opět na vstup přivedeno skokové analogové napětí u . V čase $t_1 = 14$ s je simulován vliv poruchy na systému zapnutím rezistorové zátěže $Rz1$. Dochází tak k poklesu rychlosti otáčení servopohonu. V čase $t_2 = 22$ s je vliv poruchy vypnut, viz obrázek (Obr. 3.6). Odpojení napětí ze svorek servomotoru je provedeno v čase $t_3 = 36$ s.



Obr. 3.6. Snímání vlivu poruchy na servopohonu

3.2.1.4 Snímání parametrů při reverzaci servopohonu

Průběhy snímání pochodů při reverzaci servopohonu jsou uvedeny na obrázku (Obr. 3.7). V čase $t_1 = 1$ s je provedeno skokové napětí akční veličiny u na hodnotu -6 V. V čase $t_2 = 20$ s je akční veličina změněna skokem na hodnotu napětí $+6$ V. Vypnutí simulace nastává pro čas $t_3 = 41$ s. Jako v předchozích experimentech je vzorkovací perioda T_s zachována na hodnotě $0,01$ s. Pro společné grafické porovnání charakteristik je rychlost otáčení hřídele vynesena v napěťovém rozsahu. Z průběhů snímání je patrné, že při startu a vypnutí aplikace na stejnosměrném motoru vznikají proudové špičky.

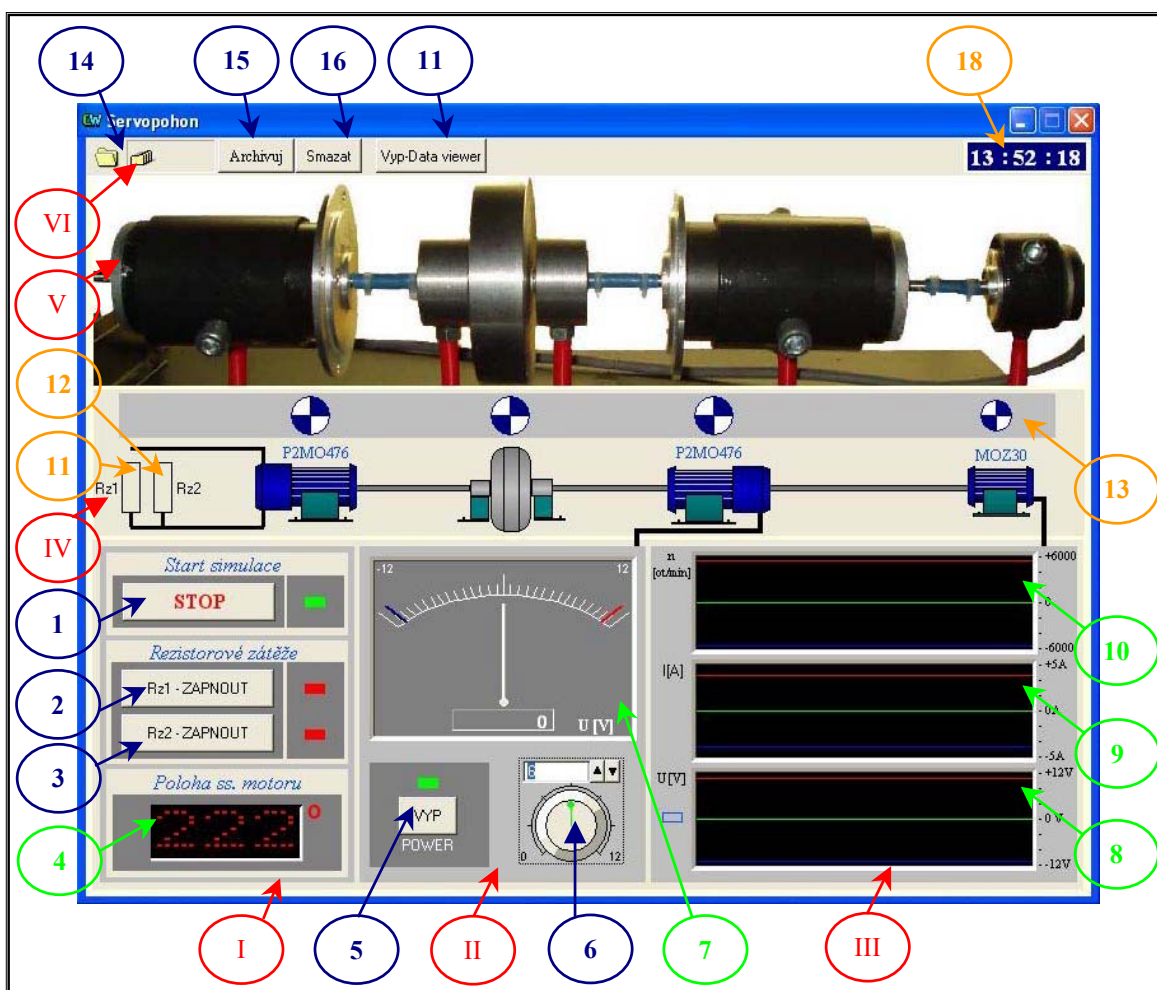


Obr. 3.7. Zobrazení dat při reverzaci modelu

4 Vizualizace ovládání modelu v Control Web 5

4.1 Úvod

Tato kapitola podává stručný přehled o softwarové části ve vývojovém prostředí *Control Web 5*. Pro snadné ovládání a informaci o stavu systému je v tomto prostředí vytvořena vizualizace, kterou zachycuje obrázek (Obr. 4.1). Zde je znázorněno hlavní ovládací okno pro řízení servopohonu. Tento program umožňuje snímání a ukládání naměřených hodnot (proud v kotvě motoru, poloha a rychlost otáčení hřídele modelu).



Obr. 4.1. Ovládací panel servopohonu

Základním stavebním kamenem pro aplikace systému *Control Web* jsou přístroje. Systém *Control Web* nabízí různé komponenty k tvorbě vizuálních aplikací (zobrazovací a ovládací prvky, alarmy, archivy, historické trendy apod.) Tyto komponenty jsou zabudovány v systému jako množina virtuálních přístrojů. Tato množina není pevně dána

a je možné ji libovolně rozšiřovat. Každý přístroj je dynamicky linkovaná knihovna detekovaná při startu systému.

4.2 Popis návrhu ovládacího panelu

Při spuštění programu je otevřeno hlavní okno aplikace. Ovládací panel je rozdělen do šesti podpanelů označených (I-VI).

(I) Základním ovládacím prostředím je panel přepínačů, na kterém se nacházejí grafické přepínací prvky a signalizace pro ovládání modelu. Obsahem těchto *switchů* je logické nastavení stavu veličin. Datovou výstupní hodnotou je typ boolean (true, false).

1. – **START/STOP** - Výstupní navržená aplikace ovládá zapnutí, vypnutí celé simulace servopohonu.

2. – **R_{z1}ZAPNOUT/R_{z1}VYPNOUT**, 3. – **R_{z2}ZAPNOUT/R_{z2}VYPNOUT** – Grafické tlačítka zapínají nezávisle na sobě rezistorové zátěže R_{z1}, R_{z2} připojené na svorky stejnosměrného motoru M2.

4. – **POLOHA SS. MOTORU** – Číselné zobrazení polohy je znázorněno komponentou *meter*, kde vstupní hodnotou je upravené elektrické napětí úměrné poloze motoru.

(II) Tento panel simuluje graficky napěťový zdroj v rozsahu ± 12 V. Je vytvořen pomocí komponenty *switch*, která ovládá zapnutí napájecího napěťového zdroje. Dále pak komponenta *control*, ta slouží pro nastavení elektrického napětí na zdroji a vizuální analogový přístroj *meter*.

5. – **POWER ZAP/VYP** – *Switch* pro zapnutí a vypnutí napájení zdroje.

6. – **NASTAVENÍ NAPĚTÍ** – Ovládací komponenta (*control*), která dává možnost nastavení číselné hodnoty výstupního elektrického napětí v rozsahu ± 12 V do datového elementu pomocí knoflíku. Výstupní veličina *controlu* je proměnná typu real.

7. – **ANALOGOVÝ UKAZATEL NAPĚTÍ** – Zobrazení číselné hodnoty elektrického napětí pomocí ručkového analogového přístroje s možností přiřazení výsledku do výstupního datového elementu. Přístroj *meter* je vyhodnocován proměnnou, která je výstupem z analogové PC karty.

(III) V pravé dolní části hlavního panelu se nachází podpanel s grafickými ukazateli:

8. – **GRAFICKÝ UKAZATEL NAPĚTÍ**

9. – **GRAFICKÝ UKAZATEL PROUDU**

10. – **GRAFICKÝ UKAZATEL OTÁČEK**

- Grafické ukazatele jsou vytvořeny komponentou s názvem *meter*, u kterých je změněn typ módu přístroje na *flow_graph*. Dvoj kliknutím myši na tuto komponentu je otevřeno samostatné panelové okno s tímto přístrojem.

(IV) Na následujícím panelu se nacházejí signalizační prvky. Signalizace pro rezistorové zátěže je tvořena přístrojem *box*. Je to jednoduchý grafický symbol (obdélník), kterému lze pomocí nativních procedur nastavovat barvu, velikost a pozici. Další signalizací je rotace motorů pomocí komponenty *engine*.

11. – Signalizace **ZAPNUTÍ/VYPNUTÍ** rezistorové zátěže Rz1, je-li podmínka viditelnosti přístroje *box* (rezistor) = true, je aktivní. Tzn., že je zapnuta odporová zátěž Rz1 na motoru M2.

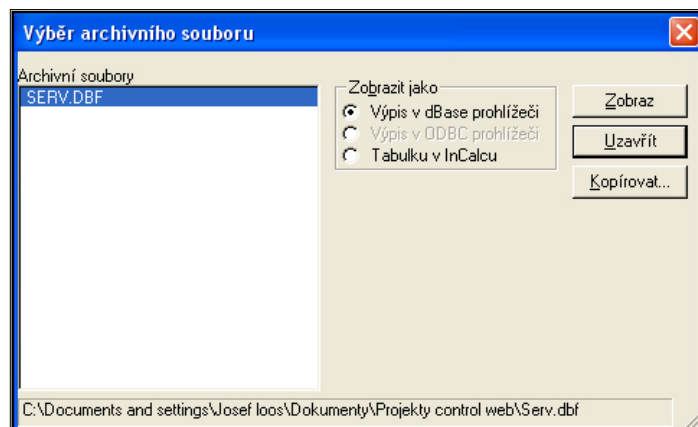
12. – Signalizace **ZAPNUTÍ/VYPNUTÍ** rezistorové zátěže Rz2, je-li podmínka viditelnosti přístroje *box* (rezistor) = true, je aktivní. Tzn., že je zapnuta odporová zátěž Rz2 na motoru M2. Obsahem viditelnosti těchto komponent je parametr logického výrazu.

13. – Signalizace **OTÁČENÍ POHONŮ** – O signalizaci otáčení motorů a inkrementálního snímače se starají komponenty zvané *engine*. Obsahem tohoto parametru je logický výraz, podle jehož výsledku bude přístroj aktivním, tj. symbol bude rotovat v závislosti na nastavení výstupního elektrického napětí. Přístroj tedy umožňuje i animaci rotace motorů ve dvou směrech otáčení.

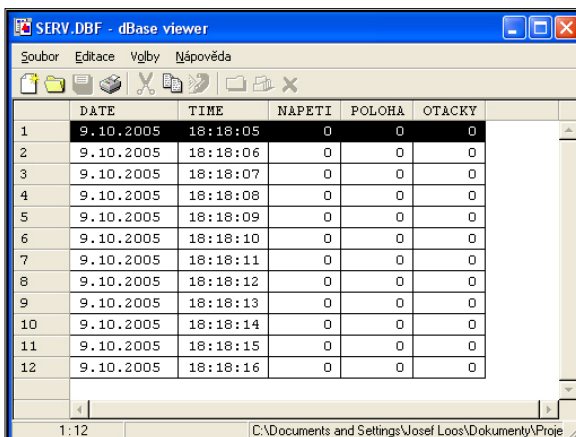
(V) Vizuální panel s obrázkem mechanického hřídelového propojení servopohonu.

(VI) Panel pro práci s naměřenými daty.

14. – **OTEVŘENÍ ARCHIVACE** – Při otevření souboru je zobrazeno panelové okno viz obrázek (Obr. 4.2). Funkce přístroje *archiver* je archivace požadovaných dat do standardního databázového souboru ve formátu **dBase** viz obrázek (Obr. 4.3), nebo v tabulce **InCalc** dle obrázku (Obr. 4.4). *DBase viewer* je jednoduchý prohlížeč dat, ve kterém není možnost žádné úpravy. Oproti tomu *InCalc viewer* umožňuje různou práci, úpravu naměřených dat. V přístroji *archiver* lze také definovat datovou historii, respektive počet souborů uchovávaných na disku nebo v databázích. Aby mohl tento přístroj archivovat data, musí být splněna podmínka pro povolení činnosti.

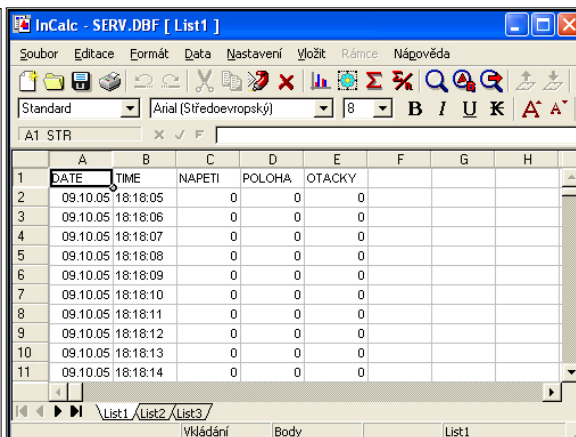


Obr. 4.2. Otevření archivace



	DATE	TIME	NAPETI	POLOHA	OTACKY
1	9.10.2005	18:18:05	0	0	0
2	9.10.2005	18:18:06	0	0	0
3	9.10.2005	18:18:07	0	0	0
4	9.10.2005	18:18:08	0	0	0
5	9.10.2005	18:18:09	0	0	0
6	9.10.2005	18:18:10	0	0	0
7	9.10.2005	18:18:11	0	0	0
8	9.10.2005	18:18:12	0	0	0
9	9.10.2005	18:18:13	0	0	0
10	9.10.2005	18:18:14	0	0	0
11	9.10.2005	18:18:15	0	0	0
12	9.10.2005	18:18:16	0	0	0

Obr. 4.3. dBase viewer



	DATE	TIME	NAPETI	POLOHA	OTACKY
1	09.10.05	18:18:05	0	0	0
2	09.10.05	18:18:06	0	0	0
3	09.10.05	18:18:07	0	0	0
4	09.10.05	18:18:08	0	0	0
5	09.10.05	18:18:09	0	0	0
6	09.10.05	18:18:10	0	0	0
7	09.10.05	18:18:11	0	0	0
8	09.10.05	18:18:12	0	0	0
9	09.10.05	18:18:13	0	0	0
10	09.10.05	18:18:14	0	0	0
11	09.10.05	18:18:15	0	0	0

Obr. 4.4. InCalc viewer

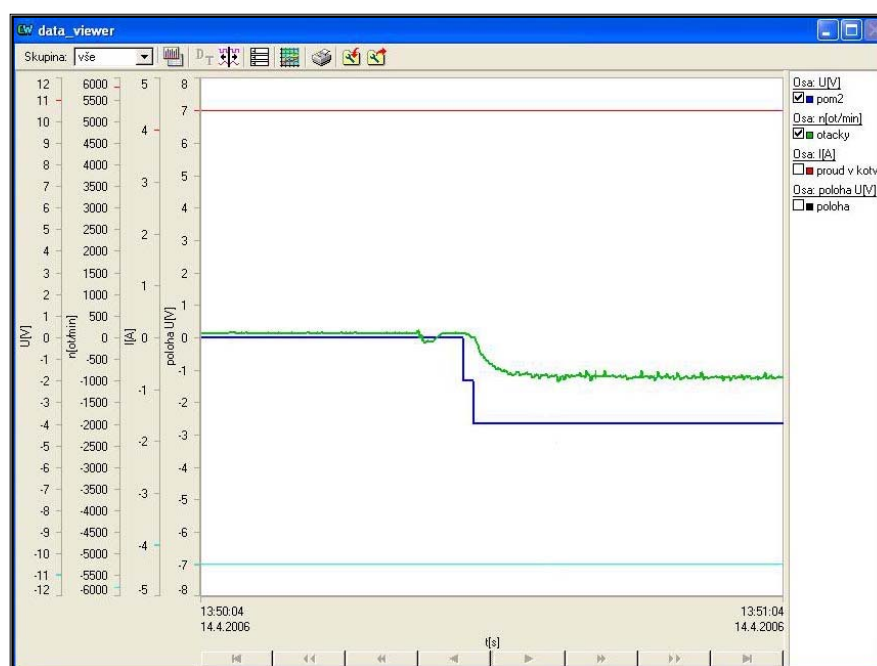
15. – **ARCHIVUJ/NEARCHIVUJ** – pomocí tohoto *switche* je zapnuta-vypnuta archivace měřených dat. Výstupním parametrem je booleovský typ, který povoluje-zakazuje činnost pro archivaci dat přístroje *archiver*.

16. – **SMAZAT** – *Switch*, který je používán pro smazání uložených archivních dat. Je-li výstupní logický parametr aktivní, je aktivní i podmínka pro smazání nastaveného permanentního souboru přístroje *archiver*.

17. – **ZAPNOUT/VYPNOUT Data viewer** – Tímto *Switchem* je otevřen-zavřen přístroj s názvem *data viewer* (Obr. 4.5). Tato komponenta je určena pro zobrazování dat uložených archivními sekcemi do databázových souborů. Umožňuje ale také sledovat datové elementy bez vlastní historie, tedy datové elementy, které nejsou archivovány v databázovém souboru. Jedná se zejména o proměnné definované v sekcích **var** uvnitř aplikační sekce **data** (globální proměnné aplikace), nebo o kanály definované v sekcích **channel**. Pro tyto datové elementy přístroj vytváří lokální datový buffer, ve kterém

uchovává určitý definovaný počet hodnot elementu. Velikost bufferu je možno nastavit v rámci definice parametrů přístroje. Tyto buffery vznikají a zanikají se spuštěním resp. ukončením aplikace. Sledované datové elementy lze uspořádat do skupin (v našem případě napětí, proud, otáčky, poloha) a libovolně volit, která skupina má být aktuálně zobrazována. Přístroj také umožňuje procházet historii dat datových elementů.

18. - **HODINY** – Časový ukazatel je vytvořen pomocí tří přístrojů *meter*, kde typ přístroje je nastaven na `text_display`. Výrazy, které jsou přístroji vyhodnocovány jsou `hour`, `minute` a `second`.



Obr. 4.5. Data Viewer

4.3 Připojení signálů modelu na kanály karty

Tab. 4.1. Zapojení signálů modelu

Vstupy		
Svorkovnice	Kanál	Signál
34	101	Poloha hřídele ss. motoru
67	100	Proud v kotvě ss. motoru
68	102	Otáčky hřídele ss. motoru
Výstupy		
58	150	Analogový výstup zapojený do modelu 0-10 V
57		AGND - Analogová zem
1		Analogový výstup zapojený do modelu +5 V
5		AGND - Analogová zem
13	300	Spínání rezistorové zátěže Rz1
47	301	Spínání rezistorové zátěže Rz1
39		AGND - Analogová zem
35		Analogový výstup zapojený do modelu +12 V

V tabulce (Tab.4.1) jsou popsány použité vstupní a výstupní signály PC karty. Tímto způsobem je karta propojena s modelem a softwarem vytvořeným v Control Web 5 a je tak možné servopohon řídit, měřit v reálném čase.

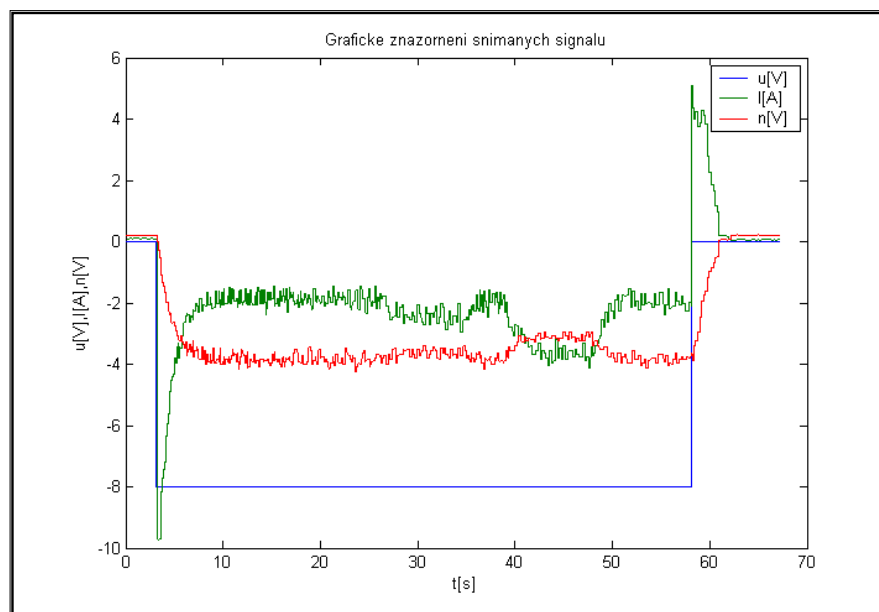
4.4 Měření na servosystému

Tab. 4.2. Měřené údaje servopohonu

bez rezistorové zátěže				rezistorová zátěž RZ1			rezistorová zátěž RZ2		
U[V]	I[A]	D/A U[V]	n[ot/min]	I[A]	D/A U[V]	n[ot/min]	I[A]	D/A U[V]	n[ot/min]
1,00	0,90	0,09	92,16	0,96	0,06	61,44	1,00	0,04	40,96
2,00	1,02	0,61	624,64	1,17	0,55	563,20	1,38	0,43	440,32
3,00	1,15	1,20	1228,80	1,44	1,05	1075,20	1,86	0,85	870,40
4,00	1,33	1,64	1679,36	1,73	1,49	1525,76	2,33	1,18	1208,32
5,00	1,45	2,23	2283,52	1,96	2,00	2048,00	2,00	1,60	1638,40
6,00	1,55	2,81	2877,44	2,21	2,48	2539,52	3,28	1,97	2017,28
7,00	1,59	3,34	3420,16	2,37	2,97	3041,28	3,80	2,35	2406,40
8,00	1,78	3,90	3993,60	2,56	3,51	3594,24	4,17	2,74	2805,76

Měření na servopohonu probíhá při postupném nastavování vstupního napětí od +1 V do +8 V. Nejdříve jsou naměřeny veličiny bez připojení rezistorových zátěží (proud v kotvě motoru M1, výstupní napětí z D/A převodníku úměrné rychlosti otáčení hřídele servopohonu a samotná rychlost otáčení modelu). V dalším kroku měření je připojena rezistorová zátěž Rz1 a k ní odečteny její příslušné měřené hodnoty. V poslední fázi měření je připojena rezistorová zátěž Rz2. Tabulka (Tab. 4.2) udává zaznamenané hodnoty měření, kde je patrné, že při sepnutí rezistorových zátěží dochází k poklesu rychlosti

otáčení a ke zvýšení proudu v kotvě stejnosměrného motoru M_1 . Na obrázku (Obr. 4.6) je graficky znázorněn jeden krok měření při vstupním napěťovém skoku ($U = -8 \text{ V}$). V čase $t = 26 \text{ s}$ je proveden zásah poruchové veličiny $Rz1$ a v čase $t = 38 \text{ s}$ zásah poruchy $Rz2$, kde je vidět patřičný pokles otáček a zvýšení proudu v kotvě stejnosměrného motoru M_1 .

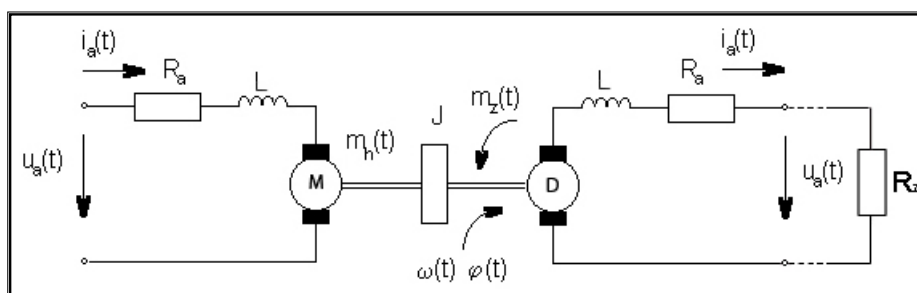


Obr. 4.6. Grafické znázornění snímaných signálů v Control Web5

5 Matematický popis, model a obrazový přenos modelu

5.1 Úvod

Abychom mohli pro model navrhnout kvalitní řízení, potřebujeme znát jeho matematický popis. Soustava má jeden vstup a několik výstupů. Na obrázku (Obr. 5.1) je v blokovém schématu znázorněn model servopohonu, kde je zanedbán zátěžový moment IRC snímače. Stejnoseměrný motor M představuje všeobecně mnoha parametrovou neautonomní nelineární soustavu se vstupním (řídícím) signálem $u_a(t)$. Přičemž výstupní signál může být jednak úhlová rychlost $\omega(t)$, moment motoru $m_h(t)$, proud obvodu kotvy $i_a(t)$, případně úhlová poloha hřídele motoru $\varphi(t)$. Zátěžný moment motoru $m_z(t)$, který vzniká působením brzdného momentu dynamu, představuje dominující poruchovou veličinu této soustavy.



Obr. 5.1. Schematické znázornění modelu

Tab. 5.1. Popis obrázku (Obr. 5.1)

Označení	Název	Jednotka
$u_a(t)$	Napětí kotvy	[V]
$i_a(t)$	Proud kotvy	[A]
R_a	Odpor kotvy	[Ω]
$m_h(t)$	Hnací moment	[N.m]
$m_z(t)$	Zátěžný moment	[N.m]
J	Celkový moment setrvačnosti	[kg.m ²]
$\omega(t)$	Úhlová rychlost	[rad.s ⁻¹]
$\varphi(t)$	Úhlové natočení	[rad]
R_z	Rezistorová zátěž	[Ω]

5.1.1 Odpor kotvy stejnosměrného motoru

Jelikož není odpor kotvy R_a stejnosměrného motoru v katalogu uveden, vypočteme ho informativně z polovičních ztrát motoru. K výpočtu použijeme nominální hodnoty stejnosměrného motoru:

$$\underline{U_n = 12 \text{ V}, P_n = 25 \text{ W}, n = 5800 \text{ ot/min}, I_n = 3,5 \text{ A} .}$$

$$\frac{U_n I_n - P_n}{2} \approx R_a \cdot I_n^2 \quad (5.1)$$

$$R_a = \frac{U_n I_n - P_n}{2 \cdot I_n^2} = \frac{12 \cdot 3,5 - 25}{2 \cdot 3,5^2} \approx 0,69 \, \Omega$$

5.1.2 Konstrukční konstanta stejnosměrného motoru

Pro výpočet konstrukční konstanty stejnosměrného motoru je výchozím vztahem rovnice:

$$C_{ss} = \frac{1}{k_m}, \quad (5.2)$$

kde k_m je zesílení stejnosměrného motoru:

$$k_m = \frac{\omega_n}{U_n - R_a I_n}. \quad (5.3)$$

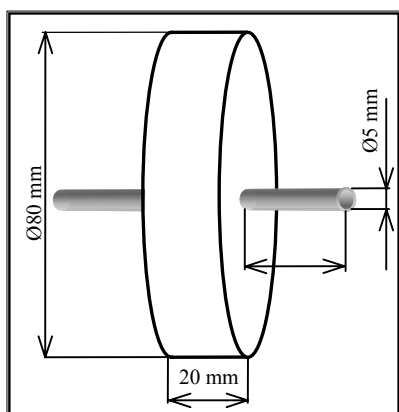
Zesílení motoru je určeno z nominálních hodnot jednotlivých veličin motoru označených indexem n. Kde U_n je nominální napětí, I_n nominální proud, R_a odpor kotvy a ω_n je nominální úhlová rychlost:

$$\omega_n = \frac{2\pi \cdot n}{60}. \quad (5.4)$$

Ve skutečnosti odpor kotvy R_a a konstrukční konstanta C_{ss} jsou veličiny částečně závislé na proudu kotvy i_a , přičemž s rostoucím proudem se tyto veličiny zmenšují. Dosazení vztahů (5.3) – (5.4) do vztahu (5.2) vypočteme konstrukční konstantu motoru:

$$C_{ss} = \frac{(U_n - R_a I_n) \cdot 60}{2\pi \cdot n} = \frac{(12 - 0,69 \cdot 3,5) \cdot 60}{2 \cdot 3,14 \cdot 5800} = 0,016 \, \text{N.m / A}.$$

5.1.3 Moment setrvačnosti modelu



Výpočet celkového momentu setrvačnosti J modelu je závislý na momentu setrvačnosti J_m obou stejnosměrných motorů, především pak na momentu setrvačnosti J_{set} setrvačníku.

$$J = J_m + J_{set} \quad (5.5)$$

Obr. 5.2. Schéma rozměrů setrvačníku

5.1.3.1 Moment setrvačnosti setrvačníku bez hřídelí

Pro výpočet momentu setrvačnosti setrvačníku J_s bez hřídelí je využito vztahu

$$J_s = \frac{1}{2} \cdot m \cdot v^2 [kg \cdot m^2], \quad (5.6)$$

kde m je hmotnost a v výška setrvačníku. Výpočet hmotnosti je dán vzorcem

$$m = \rho \cdot V, \quad (5.7)$$

kde ρ je hustota železa ($\rho_{\text{železo}} = 7870 \text{ kg} \cdot \text{m}^{-3}$) a V je objem

$$V = \pi \cdot r^2 \cdot v. \quad (5.8)$$

r je poloměr a v výška setrvačníku. Dosazením jednotlivých rozměrů do vztahů (5.7) - (5.8) je vypočítán objem a hmotnost setrvačníku:

$$V = 3,14 \cdot 0,04^2 \cdot 0,02 = 1,0048 \cdot 10^{-4} \text{ m}^3$$

$$m = 7870 \cdot 1,0048 \cdot 10^{-4} = 0,79 \text{ kg}.$$

Po dosazení hmotnosti do vztahu (5.6) je vypočítán moment setrvačnosti setrvačníku (bez hřídelí):

$$J_s = \frac{1}{2} \cdot 0,79 \cdot 0,02^2 = 1,581 \cdot 10^{-4} \text{ kg} \cdot \text{m}^2$$

5.1.3.2 Moment setrvačnosti hřídelky setrvačníku

Objem hřídelky: $V = \pi \cdot r^2 \cdot v = 5,89 \cdot 10^{-7} \text{ m}^3$

Hmotnost hřídelky: $m = \rho \cdot V = 7870 \cdot 5,89 \cdot 10^{-7} = 4,63 \cdot 10^{-3} \text{ kg}$

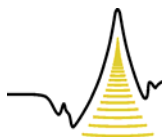
Moment setrvačnosti hřídelky: $J_h = \frac{1}{2} \cdot m \cdot v^2 = \frac{1}{2} \cdot 4,63 \cdot 10^{-3} \cdot 0,03^2 = 2,084 \cdot 10^{-6} \text{ kg} \cdot \text{m}^2$

Moment setrvačnosti celé hřídelky: $J_{2h} = 2 \cdot J_h = 2 \cdot 2,084 \cdot 10^{-6} = 4,168 \cdot 10^{-6} \text{ kg} \cdot \text{m}^2$

5.1.3.3 Celkový moment setrvačnosti setrvačníku

Výsledný moment setrvačnosti setrvačníku je součtem momentu setrvačnosti hřídelek a samotného setrvačníku:

$$J_{\text{set}} = J_s + J_{2h} = 1,6227 \cdot 10^{-4} \text{ kg} \cdot \text{m}^2.$$



Celkový moment setrvačnosti modelu je roven součtu momentů setrvačnosti ss. motorků ($J_m = 5,7799 \cdot 10^{-4} \text{ kg.m}^2$) a momentu setrvačnosti setrvačnicku J_{set} .

$$J = J_m + J_{set} = 5,7799 \cdot 10^{-4} + 1,6227 \cdot 10^{-4} = \underline{7,4026 \cdot 10^{-4} \text{ kg.m}^2}$$

5.2 Matematický popis modelu

Matematický model je matematické vyjádření podstatných vlastností existujícího systému, který popisuje znalosti o systému v použitelné formě. Matematický popis servopohonu vychází ze základních pohybových a napěťových rovnic obou stejnosměrných motorů. Základní pohybová rovnice pro stejnosměrný motor a dynamo je ve tvaru:

$$J \frac{d\omega(t)}{dt} = m_h(t) - m_z(t), \quad (5.9)$$

kde m_z je zatěžovací moment soustavy. U stejnosměrného motoru a dynamu s konstantním buzením hnací moment $m_h(t)$ je přímo úměrný proudu kotvy $i_a(t)$, proto lze psát

$$m_h(t) = C_{ss} \cdot i_a(t), \quad (5.10)$$

kde C_{ss} je momentová konstanta motoru [N.m.A^{-1}].

Použijeme-li Laplaceovou transformaci za předpokladu nulových počátečních podmínek na vztahy (5.9) - (5.10), dostáváme rovnice v obrazovém tvaru:

$$Js\Omega(s) = M_h(s) - M_z(s), \quad (5.11)$$

$$M_h(s) = C_{ss} \cdot I_a(s) \quad (5.12)$$

Pro stejnosměrný motor řízený napětím kotvy uvažujeme napěťovou rovnici motoru:

$$R_a \cdot i_a(t) + L \frac{di_a(t)}{dt} = u_a(t) - u_i(t) \quad (5.13)$$

$$u_i(t) = C_{ss} \cdot \omega(t) \quad (5.14)$$

jejíž Laplaceův obraz je ve tvaru:

$$R_a \cdot I_a(s) + LsI_a(s) = U_a(s) - C_{ss} \cdot \Omega(s). \quad (4.15)$$

Pro stejnosměrné dynamo je uvažována napěťová rovnice:

$$-R_a \cdot i_a(t) + L \frac{di_a(t)}{dt} = u_a(t) - C_{ss} \cdot \omega(t). \quad (5.16)$$

Vyjádřením proudu $i_a(t)$ z momentové rovnice dynama

$$i_a(t) = \frac{J \frac{d\omega(t)}{dt} + m_z}{C_{ss}} \quad (5.17)$$

a následným dosazením do upravené napěťové rovnice dostaneme vztah pro výsledné výstupní napětí stejnosměrného motoru M_2

$$u_a(t) = C_{ss} \cdot \omega(t) - R_a \cdot \left(\frac{J \frac{d\omega(t)}{dt} + m_z}{C_{ss}} \right) + \frac{L di_a(t)}{dt} \quad (5.18)$$

Pro výpočet zatěžovacího momentu dynama uvažujeme vztah

$$m_z(t) = \frac{P}{\omega(t)}, \quad (5.19)$$

kde P je výkon dynama

$$P = \frac{U^2}{R_z} \quad (5.20)$$

a R_z je rezistorová zátěž připojována na svorky dynama. Po dosazení vztahu (5.19) do (5.20) dostaneme výsledný zatěžovací moment:

$$m_z(t) = \frac{U^2}{R_z \cdot \omega(t)} \quad (5.21)$$

Celkový brzdný moment M je pak tvořen součtem zatěžovacího momentu M_z a vlastních brzdných momentů stejnosměrných motorů ΔM_m :

$$M = M_z + \Delta M_m, \quad (5.22)$$

kde ΔM_m je rozdíl nominálního elektromagnetického momentu M_{mn} a nominálního momentu motoru na hřídeli:

$$\Delta M_m = M_{mn} - M_{mh}. \quad (5.23)$$

Výpočet nominálního elektromagnetického momentu dle vztahu:

$$M_{mn} = C_{ss} \cdot I_n = 0,016 \cdot 3,5 = \underline{56 \text{ mN.m}}. \quad (5.24)$$

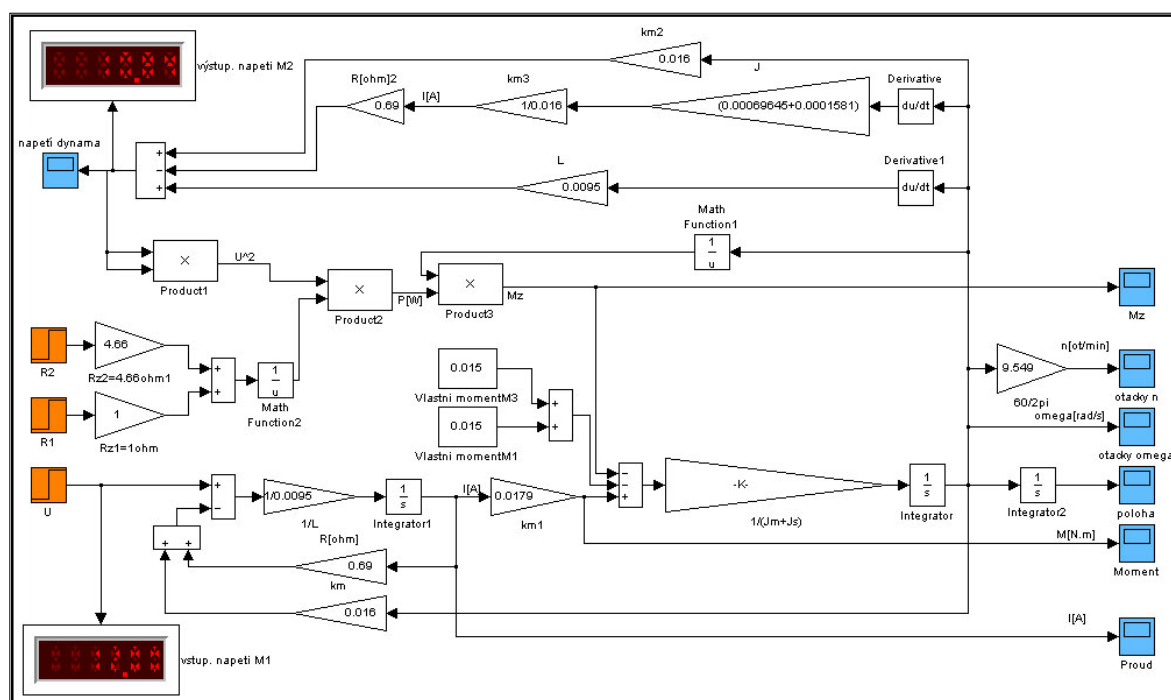
Výpočet nominálního momentu motoru na hřídeli dle vztahu:

$$M_{mh} = \frac{P_n}{\omega_n} = \frac{25}{607} = \underline{41 \text{ mN.m}}. \quad (5.25)$$

Dosažením vztahů (5.24) – (5.25) do vztahu (5.23) dostaneme výsledný vlastní brzdný moment ss. motoru:

$$\Delta M_m = (C_{ss} \cdot I_n) - \frac{P_n}{\omega_n} = 56 \cdot 10^{-3} - 41 \cdot 10^{-3} = 15 \text{ mN.m} \quad (5.26)$$

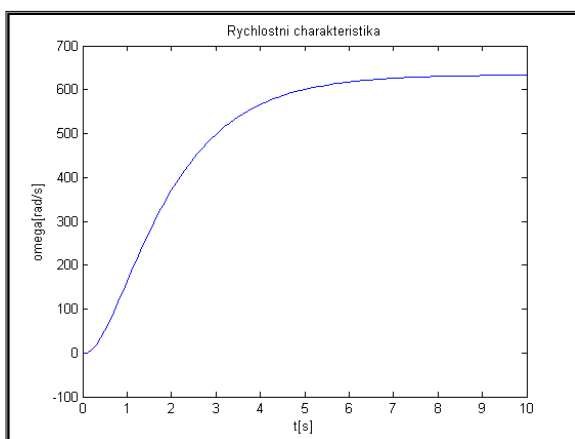
Dle vztahů (5.11), (5.12), (5.15), (5.18), (5.21) a (5.26) je sestaveno blokové schéma modelu servopohonu v softwaru MATLAB Simulink viz obrázek (Obr. 5.3).



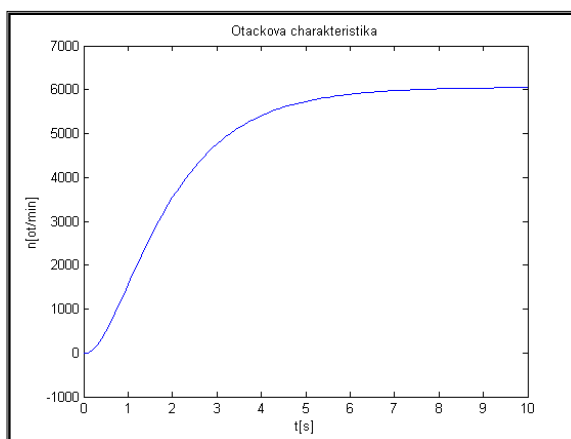
Obr. 5.3. Blokové schéma modelu servopohonu

5.2.1 Simulace modelu pomocí matematického popisu

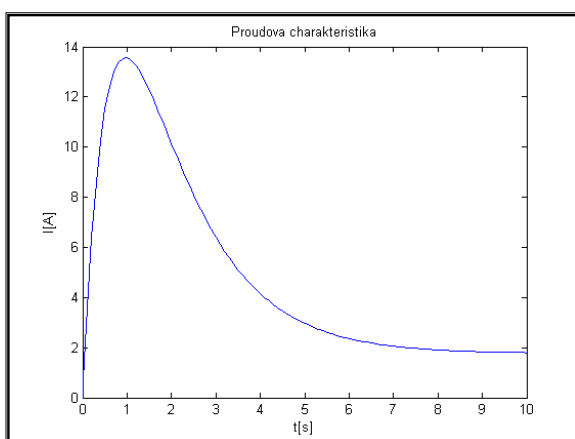
Simulace modelu podle obrázku (Obr.5.3) probíhá při vstupním napětí $U = +12\text{ V}$. Rezistorové zátěže R_z nejsou na svorky motoru M_2 připojeny, tím nepůsobí na hnací stejnosměrný motor M_1 zátěžný moment $M_z = 0$. Výsledné výstupní závislosti jsou znázorněny na následujících obrázcích.



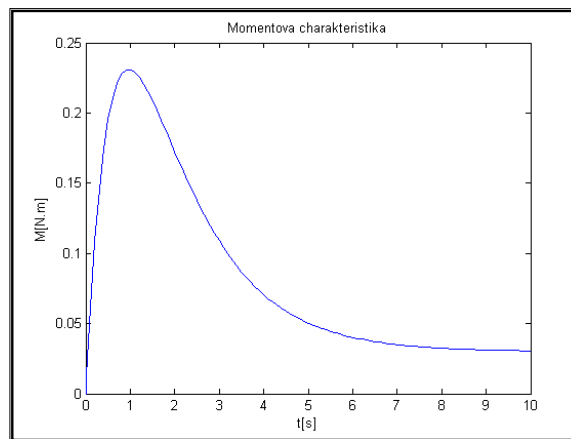
Obr. 5.4. Závislost rychlosti na čase t



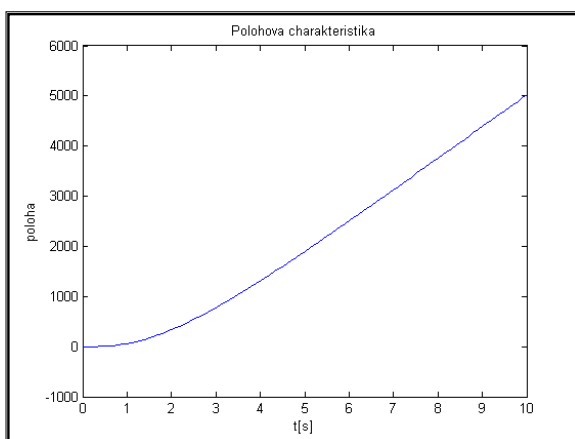
Obr. 5.5. Závislost otáček na čase t



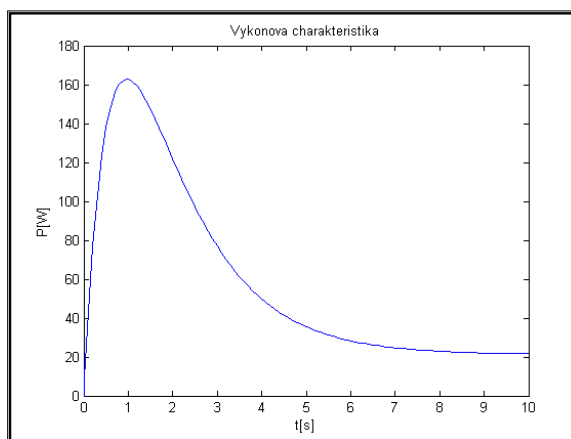
Obr. 5.6. Závislost proudu na čase t



Obr. 5.7. Závislost momentu na čase t



Obr. 5.8. Závislost polohy na čase t



Obr. 5.9. Závislost výkonu na čase t

5.3 Obrazový přenos modelu

Obrazový přenos modelu vychází z dynamických stavů stejnosměrných motorů. Ze základních pohybových rovnic motoru (5.9) a (5.10) vyjádříme rovnici proudu kotvy i_a :

$$i_a(t) = \frac{J}{C_{ss}} \frac{d\omega(t)}{dt} + \frac{m_z(t)}{C_{ss}}. \quad (5.24)$$

Tento vztah dosadíme do napěťové rovnice (5.13) a matematicky upravíme:

$$\begin{aligned} u_a(t) &= R_a \left[\frac{J}{C_{ss}} \frac{d\omega(t)}{dt} + \frac{m_z(t)}{C_{ss}} \right] + L_a \left[\frac{J}{C_{ss}} \frac{d^2\omega(t)}{dt^2} + \frac{1}{C_{ss}} \frac{dm_z(t)}{dt} \right] + C_{ss} \cdot \omega(t), \\ u_a(t) &= \frac{R_a J}{C_{ss}} \frac{d\omega(t)}{dt} + \frac{R_a m_z(t)}{C_{ss}} + \frac{L_a J}{C_{ss}} \frac{d^2\omega(t)}{dt^2} + \frac{L_a}{C_{ss}} \frac{dm_z(t)}{dt} + C_{ss} \cdot \omega(t) \cdot \frac{1}{C_{ss}}, \\ u_a(t) \cdot \frac{1}{C_{ss}} &= \frac{R_a J}{C_{ss}^2} \frac{d\omega(t)}{dt} + \frac{R_a m_z(t)}{C_{ss}^2} + \frac{L_a J}{C_{ss}^2} \frac{d^2\omega(t)}{dt^2} + \frac{L_a}{C_{ss}^2} \frac{dm_z(t)}{dt} + \omega(t). \end{aligned} \quad (5.25)$$

Provedeme-li Laplaceovu transformaci na vztah (5.25), dostaneme obrazový tvar rovnice:

$$U_a(s) \cdot \frac{1}{C_{ss}} = \frac{R_a J}{C_{ss}^2} s \Omega(s) + \frac{L_a J}{C_{ss}^2} s^2 \Omega(s) + \frac{R_a}{C_{ss}^2} M_z(s) + \frac{L_a}{C_{ss}^2} s M_z(s), \quad (5.26)$$

Označíme-li:

$$\tau_a = \frac{L_a}{R_a}, \quad \tau_m = \frac{R_a J}{C_{ss}^2}, \quad (5.27), (5.28)$$

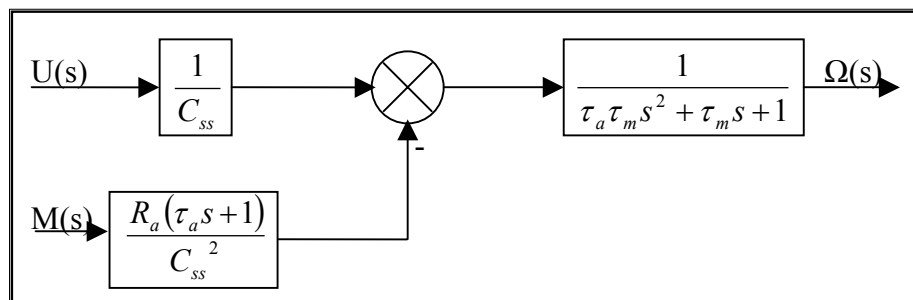
kde τ_a je elektromagnetická časová konstanta a τ_m elektromechanická časová konstanta.

Dosadíme-li vztahy τ_a , τ_m do vztahu (5.26) dostaneme výsledný přenos ss. motoru:

$$\Omega(s) = \frac{1}{C_{ss}} \frac{1}{\tau_a \tau_m s^2 + \tau_m s + 1} \cdot U(s) - \frac{R_a}{C_{ss}^2} \frac{\tau_a s + 1}{\tau_a \tau_m s^2 + \tau_m s + 1} \cdot M_z(s). \quad (5.29)$$

Napětí kotvy zde hraje roli akční a zatěžovací moment poruchové veličiny. Pak je možné zapsat přenos regulované soustavy $G_s(s)$ a přenos poruchové veličiny $G_d(s)$ do vztahů:

$$G_s(s) = \frac{1}{C_{ss}} \frac{1}{\tau_a \tau_m s^2 + \tau_m s + 1}, \quad G_d(s) = -\frac{R_a}{C_{ss}^2} \frac{\tau_a s + 1}{\tau_a \tau_m s^2 + \tau_m s + 1}. \quad (5.30), (5.31)$$



Obr. 5.10. Blokové schéma obrazových přenosů

5.3.1 Identifikace modelu pomocí obrazového přenosu

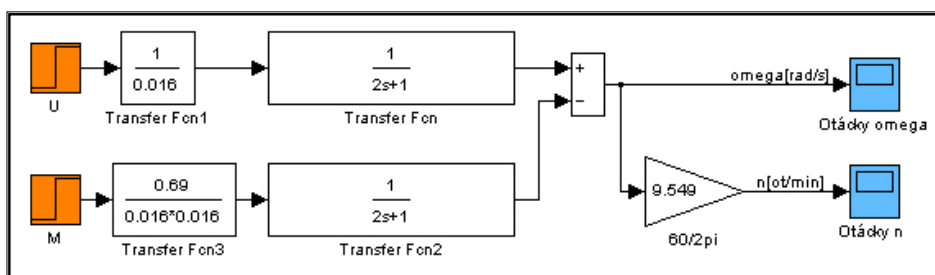
Jelikož se model chová jako soustava prvního řádu, uvažujme pro výpočet časových konstant nulovou indukčnost ss. motoru $L = 0$ H. Odpor kotvy je z předchozích výpočtů roven $R_a = 0,69 \Omega$ a konstanta motoru $C_{ss} = 0,016$ N.m/A.

$$\tau_a = \frac{L_a}{R_a} = \frac{0}{0,69} = 0 \quad ; \quad \tau_m = \frac{R_a J}{C_{ss}^2} = \frac{0,69 \cdot 0,74026 \cdot 10^{-3}}{0,016^2} = 2.$$

Dosažením konstant $\tau_a = 0$, $\tau_m = 2$ do vztahů (5.30, 5.31) dostaneme model dynamického systému prvního řádu s poruchovou veličinou:

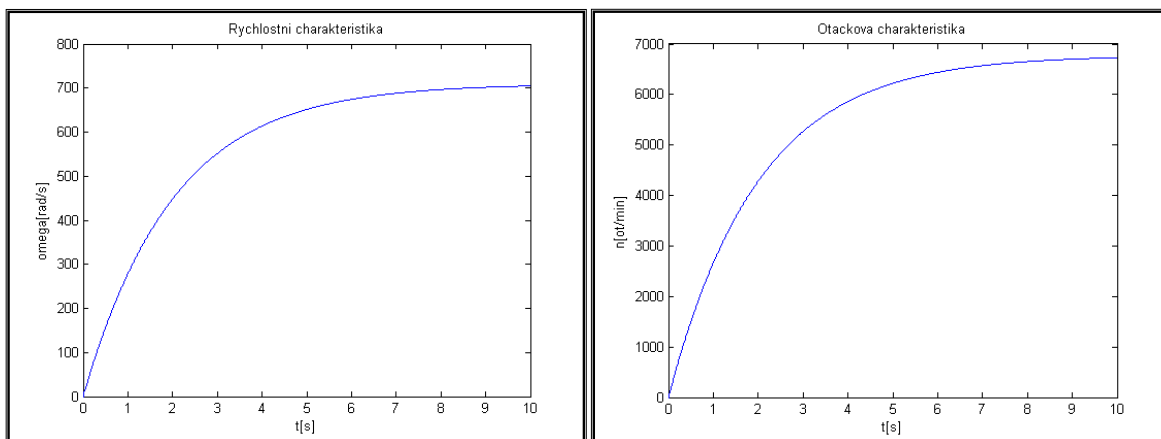
$$G_s(s) = \frac{1}{0,016} \frac{1}{2s+1}, \quad G_d(s) = -\frac{0,69}{0,016^2} \frac{1}{2s+1}.$$

Struktura modelu dynamického systému s účinkem poruchové veličiny je znázorněna na obrázku (Obr. 5.11).



Obr. 5.11. Model dynamického systému servopohonu

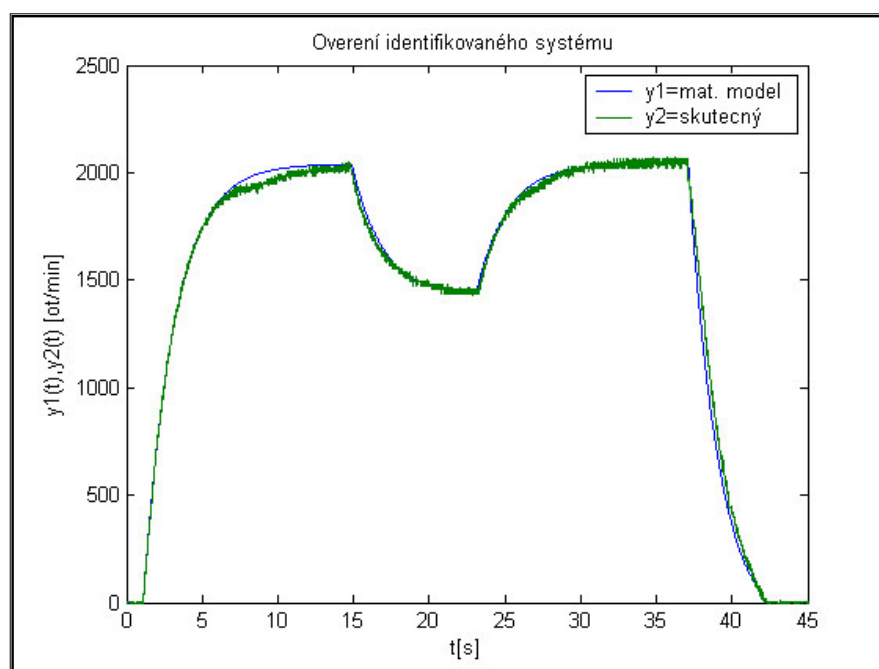
Pro identifikační simulaci je zvolena akční veličina napětí $U = 12$ V. Zatěžovací moment M_z je nulový tzn., že není připojena rezistorová zátěž na svorky dynama. Na obrázcích (Obr. 5.12, Obr. 5.13) je zakreslena odezva soustavy servopohonu na akční veličinu.



Obr. 5.12. Graf rychlosti v závislosti na čase t **Obr. 5.13.** Graf otáček v závislosti na čase t

5.4 Ověření identifikovaného systému

Ověření, zda model odpovídá identifikovanému systému, je provedeno porovnáním odezvy skutečného systému a modelu obrazového přenosu servopohonu.



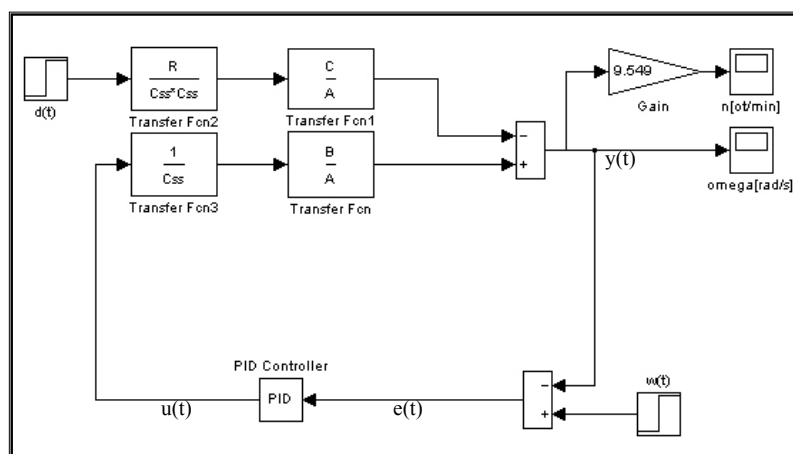
Obr. 5.14. Ověření identifikovaného systému

V tomto identifikačním experimentu je na vstup přivedeno opět skokové analogové napětí. Pomocí rezistorové zátěže R_{z1} je na systému v čase $t_1 = 15$ s simulována porucha. V čase $t_2 = 23$ s je vliv poruchy vypnut. Porovnání výsledných charakteristik je na obrázku (Obr. 5.14).

6 Regulační experimenty modelu pomocí obrazových přenosů

6.1 Úvod regulace

Cílem regulace je kompenzace vlivů poruchových veličin, především poruchové veličiny tvořené brzdícím momentem pomocí zatěžovacích rezistorů. Dále pak dosažení požadovaných dynamických vlastností obvodu a hodnot regulované veličiny. Pro regulaci je zvoleno zpětnovazební řízení, které využívá informaci o účinku řízení a poruch na výstupu regulované soustavy. Tyto informace jsou obsaženy v regulační odchylce $e(t)$, která je vstupem do regulátoru. Žádanou hodnotou $w(t)$ jsou otáčky servopohonu a poruchu $d(t)$ tvoří momentová zátěž viz obrázek (Obr. 6.1).



Obr. 6.1. Zpětnovazební řízení servopohonu

Model pro regulaci vychází z identifikace obrazových přenosů servopohonu:

$$G_s(s) = \frac{1}{0,016} \frac{1}{2s+1}, \quad G_d(s) = -\frac{0,69}{0,016^2} \frac{1}{2s+1}$$

Struktura obrazových přenosů je definována vztahy

$$G_s(s) = \frac{1}{C_{ss}} \cdot \frac{B(s)}{A(s)}; \quad G_d(s) = \frac{R_a}{C_{ss}^2} \cdot \frac{C(s)}{A(s)}, \quad (6.1), (6.2)$$

kde je $A(s)$ polynom jmenovatele, $B(s)$ a $C(s)$ jsou polynomy čitatele.

6.1.1 Obrazový přenos regulátoru

Nejčastěji používaným regulátorem v regulačních schématech je regulátor typu PID (proporcionálně integračně derivační regulátor), jehož vstupem je regulační odchylka a výstup tvoří vážený součet z regulační odchylky, jejího integrálu a derivace. Obrazový přenos regulátoru je možno vyjádřit ve tvaru

$$R(s) = K_R \left[1 + \frac{1}{T_I s} + T_D s \right] = r_0 + \frac{r_1}{s} + r_2 s, \quad (6.3)$$

kde je K_R ... proporcionální zesílení všech složek regulátoru,

T_I ... integrační časová konstanta,

T_D ... derivační časová konstanta,

$r_0 = K_R$... proporcionální zesílení,

$r_1 = K_R/T_I$... proporcionální zesílení integrační složky,

$r_2 = K_R \cdot T_D$... proporcionální zesílení derivační složky.

Jedná se tedy o paralelní (standardní) tvar regulátoru. Obraz z výstupu regulátoru je $U(s) = R(s)E(s)$, kde $E(s)$ je obraz regulační odchylky.

6.2 *Optimální časové seřízení PID regulátoru*

Uvažujme opět zpětnovazební obvod regulované soustavy dle obrázku (Obr. 6.1). Do soustavy vstupuje poruchová veličina $d(t)$ a žádaná hodnota $w(t)$. Přenos regulátoru je ve tvaru

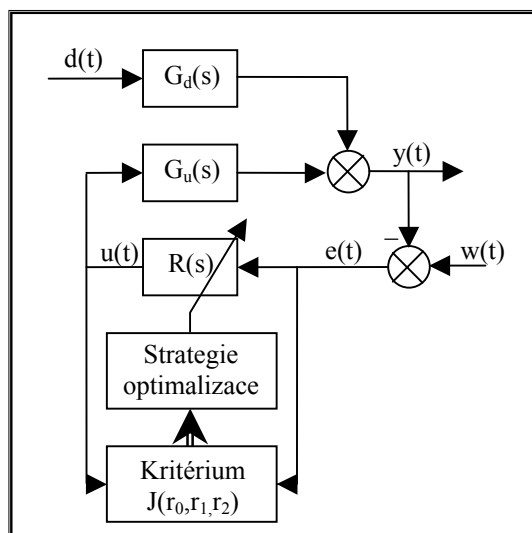
$$R(s) = r_0 + \frac{r_1}{s} + \frac{r_2 s}{T_v s + 1} = P + \frac{I}{s} + \frac{Ds}{Ns + 1}, \quad (6.4)$$

kde je r_0 ... proporcionální zesílení, r_1 ... integrační zesílení, r_2 ... derivační zesílení,

T_v ... parazitní časová konstanta.

Pro řešení optimalizační úlohy seřízení PID regulátoru opět využijeme softwarovou podporu v prostředí MATLABu. Regulovanou soustavu a kritérium je modelováno pomocí bloků v SIMULINKu.

Optimalizace je realizována ideovým schématem dle obrázku (Obr. 6.2).



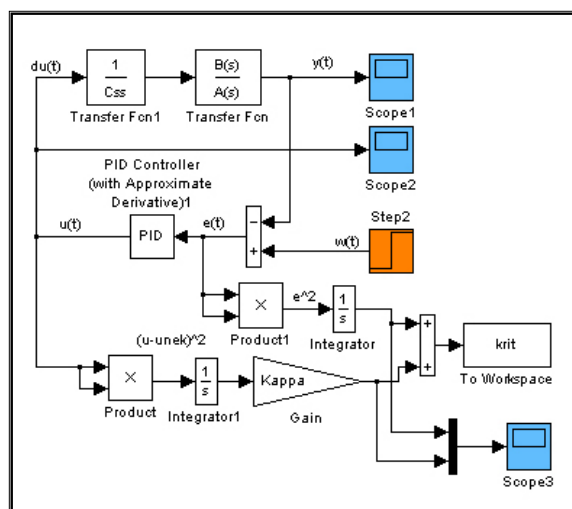
Obr. 6.2. Optimalizace parametrů PID regulátoru

Softwarový program pro optimalizaci parametrů PID regulátoru s přenosem $R(s)$ je realizován podle zobecněného kvadratického kritéria

$$J(r_0, r_1, r_2) = \int_0^{T_{sim}} \left\{ \bar{e}(t)^2 + \kappa [u(t) - u(\infty)]^2 \right\} dt \longrightarrow MINIMUM, \quad (6.5)$$

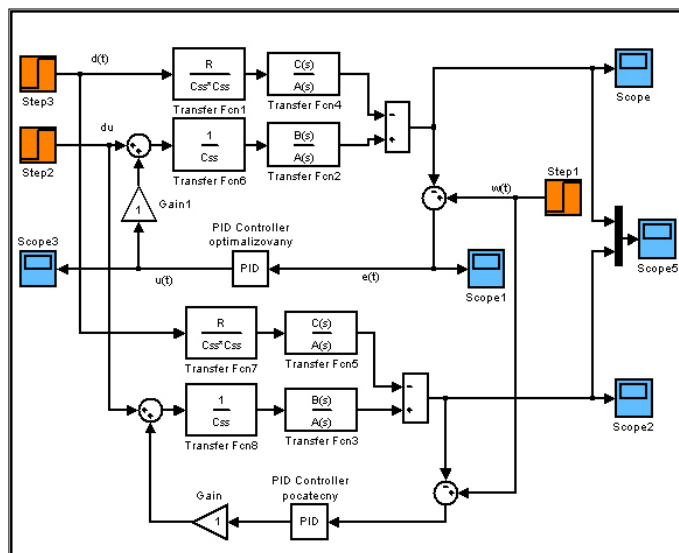
kde je $e(t)$... regulační odchylka, $u(t)$... akční veličina, $u(\infty)$... ustálená hodnota akční veličiny, κ ... váhový koeficient, jehož volbou se dosahuje požadovaného tlumení akční veličiny, r_0, r_1, r_2 ... parametry regulátoru.

Výpis programů pro optimalizaci parametrů PID regulátoru je uveden v příloze viz příloha A.4. Model uzavřeného obvodu s PID regulátorem a s výpočtem hodnoty kritéria „*krit*“ v SIMULINKu má název „*pidkra*“ a je uveden na obrázku (Obr. 6.3).



Obr. 6.3. Model pro výpočet kritéria (soubor *pidkra.mdl*)

Porovnání průběhů regulačních pochodů po optimalizaci parametrů PID regulátoru s počátečním nastavením je realizováno v SIMULINKu vytvořeným schématem PIDsima, které je uvedeno na obrázku (Obr. 6.4).



Obr. 6.4. Model pro porovnání regulačních pochodů (soubor pidsima.mdl)

Výstupem optimalizačního programu je hodnota kvadratického kritéria pro počáteční nastavení složek PID regulátoru ($P0 = 5$; $I0 = 8$; $D0 = 0.1$ – ručně nastavené parametry):

***krit* = 0,1295.**

Dále pak optimalizované parametry regulátoru PID:

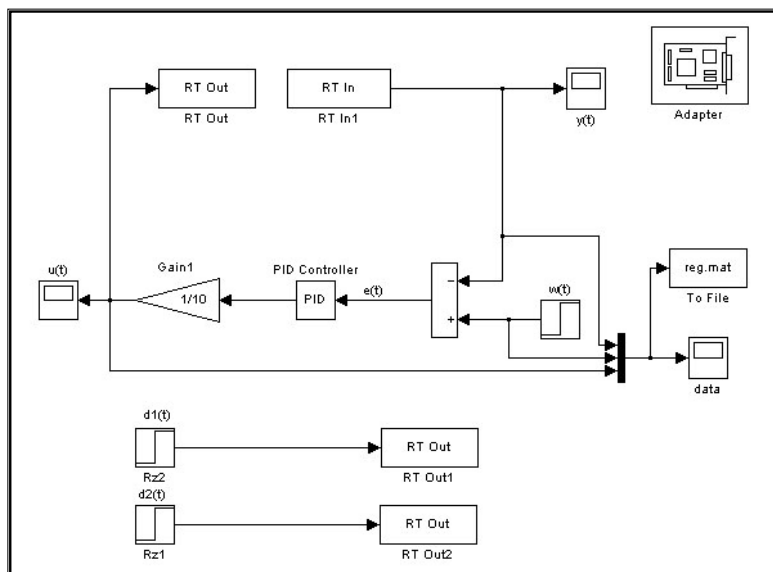
$P = 5,98$; $I = 10,20$; $D = 0,07$

a hodnota kvadratického kritéria pro optimalizované nastavení PID regulátoru:

$krit = 0,0262$.

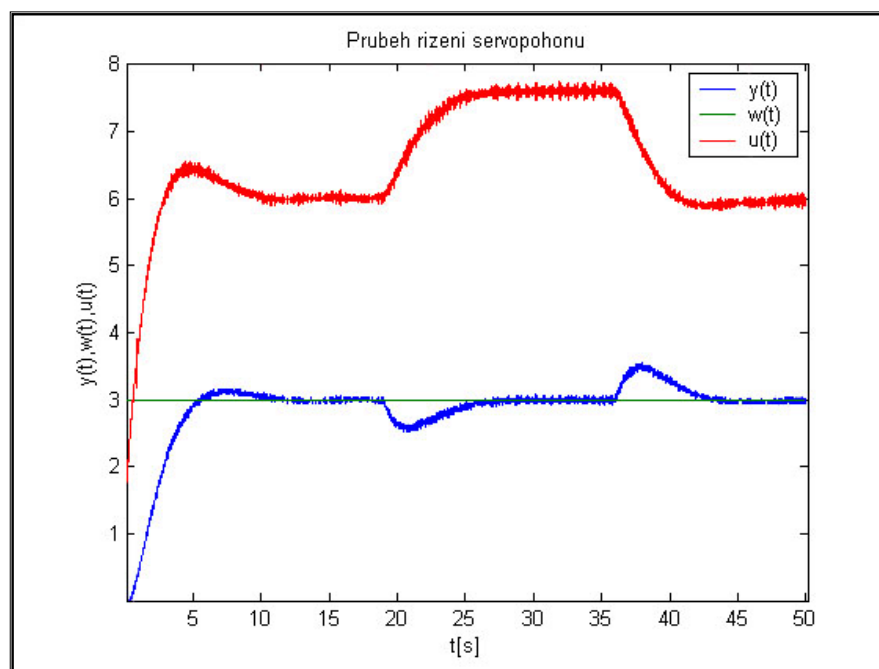
Při optimalizaci parametrů PID regulátoru hraje důležitou roli velikost váhového koeficientu Kappa, jehož velikostí se určuje tlumení akční veličiny. Velikost tohoto koeficientu je použita v desetinách.

6.2.1 Průběh optimálního řízení na reálném modelu



Obr. 6.5 Zpětnovazební obvod pro reálné řízení

Řízení servopohonového modelu je aplikováno v reálném čase dle obrázku (Obr. 6.5). Žádanou veličinou je měřená rychlost otáčení hřídele. Akční veličina je napětí na svorkách stejnosměrného motoru M_1 . Výstupní průběhy zpětnovazebního reálného řízení jsou uvedeny na obrázku (Obr. 6.6), kde je zanesena ilustrace akční $u(t)$, žádané $w(t)$ a výstupní veličiny $y(t)$. Žádaná veličina je nastavena na hodnotu 3 V, což v praxi znamená cca 3000 ot/min viz tabulka (Tab. 4.2). V čase $t_l = 19$ s je zapnuta porucha ve formě



Obr. 6.6. Regulované veličiny $y_o(t)$, $y_p(t)$

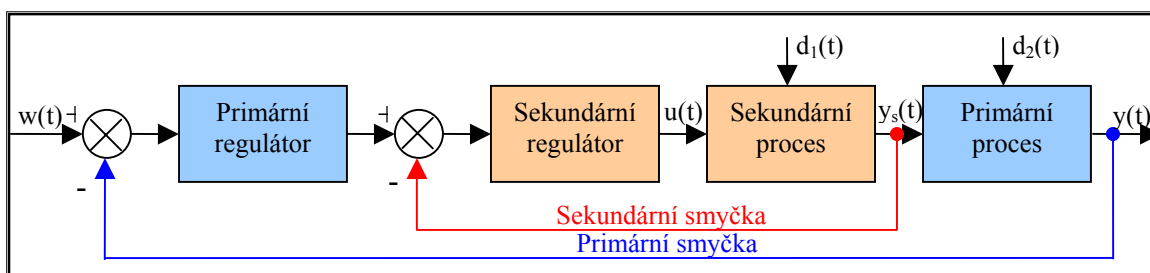
rezistorové zátěže R_{z1} . Z průběhů je patrné, že potlačení poruchy optimálně nastaveným regulátorem PID je za čas $t = 5$ s. V čase $t_2 = 36$ s je rezistorová zátěž odpojena. Perioda vzorkování pro snímání a generování veličin je nastavena na hodnotu 0,01 s.

6.3 Kaskádní regulace

Kaskádní regulace je užitečná a přínosná, můžeme-li regulovaný proces rozdělit na dvě samostatné části. První část je s rychlejší dynamikou a druhá část s pomalejší či s dopravním zpožděním. Základním předpokladem pro kaskádní regulaci je, že můžeme měřit výstupy obou částí regulovaného procesu. V našem případě proud v kotvě stejnosměrného motoru a měřená rychlost otáčení na hřídeli.

První část regulovaného procesu má menší časovou konstantu oproti druhé části. Samozřejmě, že pro takový proces je možné použít klasický jednosmyčkový regulační obvod s PID regulátorem, který byl ukázán v předchozí regulační metodě. Kaskádní uspořádání však způsobí podstatně rychlejší a kvalitnější potlačení vlivu poruch působící na jednu, nebo obě části procesu. Jednoduše řečeno v kaskádní regulaci využijeme obě měřené veličiny jednoho procesu, což představuje větší informaci, než při klasické regulaci.

Použití větší informace se projeví ve zlepšení kvality regulace. Kaskádní regulační obvod tvoří dvě smyčky: vnější (primární) smyčka a vnitřní (sekundární) smyčka. Blokové schéma kaskádní regulace je uvedeno na obrázku (Obr. 6.6).



Obr. 6.6. Schéma kaskádní regulace

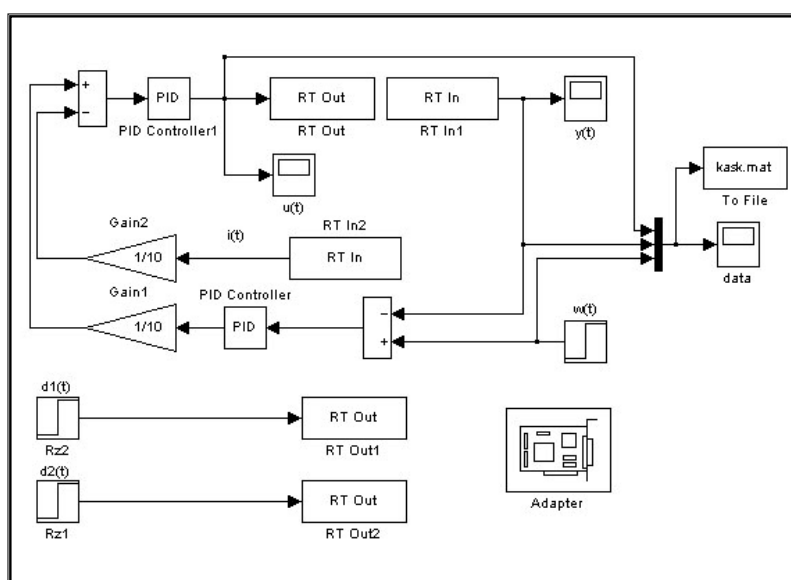
Základní výhody dvousmyčkové kaskádní regulace oproti klasické jednosmyčkové regulaci jsou tři.

První výhodou je, že poruchu, která působí na sekundární proces, potlačí sekundární regulátor před tím, než začne ovlivňovat primární proces. Účinek na primární regulovanou veličinu tak bude minimální.

Druhou výhodou je, že se urychlí reakce sekundárního procesu. Časová konstanta přenosu sekundární smyčky musí být tedy menší, než u klasické jednoduché regulační smyčky. Tím se zvýší i rychlost odezvy v primární smyčce. Jako třetí důležitá výhoda je, že sekundární regulátor tlumí změny zesílení sekundárního procesu a tak zlepšuje celkovou robustnost regulace. Z výše uvedených tří bodů je patrné zlepšení kaskádního způsobu regulace oproti klasickému jednosmyčkovému uspořádání. V případě dvousmyčkové kaskádní regulace je zejména vhodné, když v sekundárním obvodu působí hlavní poruchy celého regulačního obvodu, neboť je možné poměrně rychle a efektivně potlačit jejich vliv. Je rovněž vhodné, je-li sekundární smyčka rychlejší než primární.

Další klíčovou otázkou je, jaký typ regulátoru zvolit v sekundárním obvodu a jaký v primárním obvodu. Integrační část PID regulátoru obvykle obsahuje pouze primární regulátor. Proporcionální zesílení sekundárního regulátoru totiž bývá větší vzhledem k možnosti potlačit vliv poruch působících na sekundární obvod. Sekundárním regulátorem je proto obvyklé pouze P regulátor s větším zesílením. Primární regulátor je obvykle PI, nebo PID regulátor. Integrační část primárního regulátoru bude potlačovat vliv poruch působících nejen v sekundárním, ale také v primárním obvodu. Derivační část v souladu s jejím příznivým působením odezvu systému urychlí a stabilizuje.

Za těchto podmínek pro kaskádní řízení je sestaven v softwaru MATLAB blokový regulační obvod pro reálné řízení servopohonu viz obrázek (Obr. 6.7).



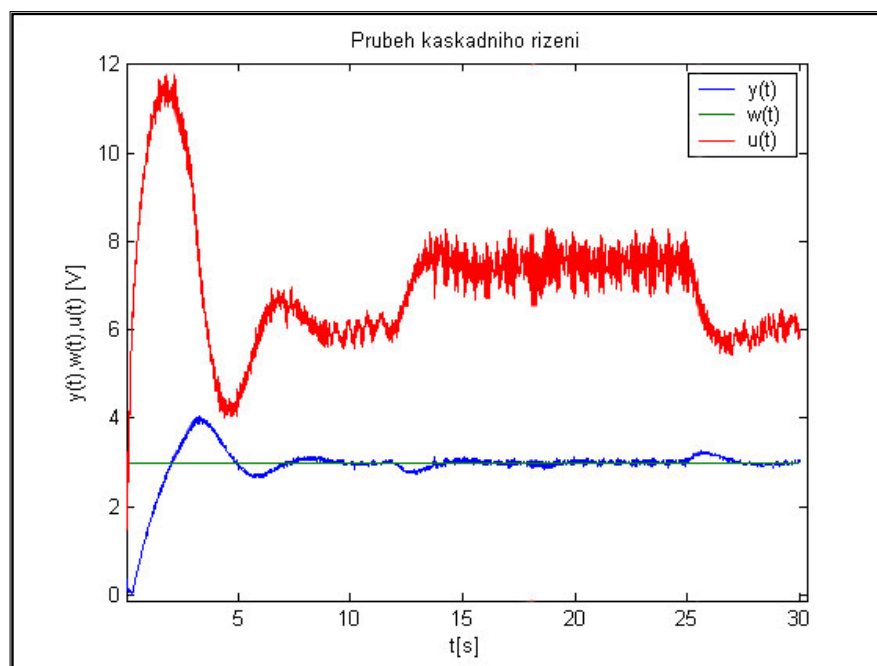
Obr. 6.7. Blokové znázornění kaskádního řízení servopohonu

Primární smyčku tvoří zpětná otáčková a sekundární smyčku zpětná proudová vazba servopohonu. Pro primární otáčkovou smyčku je zvolen regulátor typu PI a pro

sekundární proudovou smyčku regulátor typu P. Nastavení složek regulátorů je podle „ruční optimalizace“. Složky PI regulátoru ($r_0 = 2$, $r_1 = 9$).

Jak už bylo řečeno pro sekundární proudovou smyčku je zvolen pouze regulátor s proporcionální složkou s větším zesílení $r_0 = 5$.

Výsledné seřízení kaskádní regulace na žádanou hodnotu $w(t) = 3$ je na obrázku (Obr. 6.8).



Obr. 6.8. *Průběh kaskádní regulace*

Z obrázku (Obr. 6.8) je vidět, že doba ustálení na akční veličinu $w = 3$ při kaskádním řízení je menší, než-li při klasické jednosmyčkové regulaci. Také je vidět, že kaskádní regulace má o dost lepší potlačení poruchové veličiny $d(t)$, která působí na soustavu v čase $t = 12$ s. Za cenu překmitu dosáhneme rychlejšího náběhu, což je v některých případech nežádoucí.

7 Závěr

V této diplomové práci jsem navázal na můj ročníkový projekt, kde jsem dokončil řízení malého laboratorního servopohonu.

Práce popisuje návrh a vývoj modelu použitelného pro řízení a regulaci ve školních experimentech. Vytvořený model servopohonu splňuje všechny podmínky, které byly stanoveny před návrhem.

Jak jsem již na začátku předeslal, mým hlavním hardwarovým cílem bylo vyřešit a upravit snímání polohy a rychlosti otáčení hřídele servopohonu. Původní snímání signálů inkrementálního snímače pomocí multifunkční PC karty MF604 bylo nahrazeno novým elektronickým zapojením. V úvahu pro návrh snímací jednotky lze brát i vlastní samostatnou studii jednotlivých elektronických komponent, která zabrala také nezištnou část práce.

Nevýhodou inkrementálního snímače je ztráta údaje polohy při odpojení napájecího napětí. Proto se v praxi také používají absolutní snímače polohy, které mají na svém kotouči několik stop se značkami, tvořící určitý kód. Tyto čidla poskytují správný údaj i po předchozí ztrátě napájecího napětí. Tato výhoda oproti inkrementálním snímačům se značně projeví na ceně čidla.

Programování mikropočítače AT89S51 je prováděno ve strojově orientovaném jazyku assembler. Zdrojový text v assembleru zpracovává překladač a pomocí ISP programování přes paralelní port počítače je generován do paměti jednočipu.

Osobní počítač je použit jednak jako řídicí systém a jednak zajišťuje plnohodnotnou vizualizaci ovládání. Při řízení je ke komunikaci s modelem použita měřicí karta **PCI-1711** ovládaná z programu vytvořeném v Control Web 5, nebo v softwaru Matlab. V Matlabu je pro přístup ke kartě použit Real-Time toolbox. Pro monitorování stavu modelu je využito grafického rozhraní Simulink.

Na závěr jsem provedl základní identifikaci modelu pomocí matematické analýzy, při které jsem získal přenos prvního řádu. Podle získané identifikace jsem prováděl návrh regulátorů, kde jednotlivé složky regulátoru byly hledány optimalizační metodou dle zobecněného kvadratického kritéria. Navržený způsob řízení byl ověřen na reálném systému v programovém prostředí Matlab/Simulink a to, jak pro jednosmyčkové, tak i pro dvousmyčkové kaskádní řízení.

Model je tak připraven pro školní experimenty založené na regulaci a řízení

servopohonu. Vhodným námětem pro další práci je připojení k reálnému regulátoru a vytvořit tak kompaktní řízený servopohon.

Výhodou navrženého modelu jsou jeho malé rozměry a jeho snadná přenosná manipulace. V budoucnu by bylo vhodné upravit připevnění motorů a setrvačníku, jelikož při velkých otáčkách dochází k vibracím a to může narušit řízení servomechanismu.

Využití snímání polohy a rychlosti otáčení hřídele motoru je současným trendem v dnešním vývoji, kde předmětem některých firem je stavba a konstrukce poloautomatických strojů pro odstranění manuální lidské práce.

Literatura

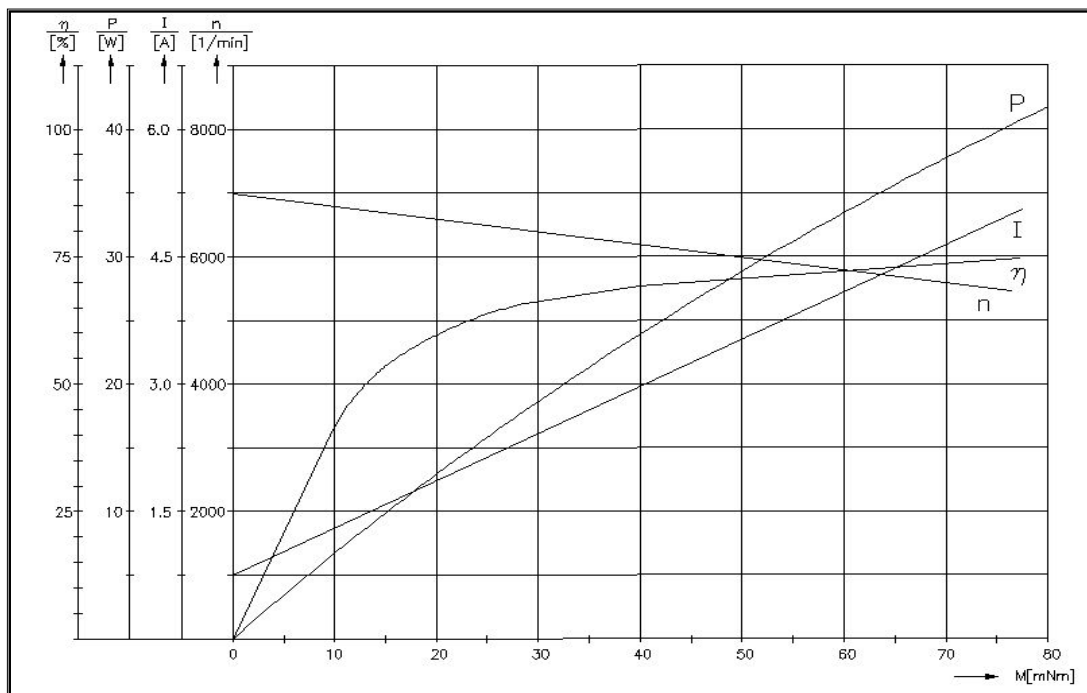
- [1] BABÁK M., LAURINOVÁ V.:
Programovací jazyk assembler 8051. Tesla Eletos, Praha 1987.
- [2] BABÁK M., CHLÁDEK L.:
Architektura a technické vlastnosti jednočipových mikrořadičů 8051. Tesla Eletos, Praha 1987.
- [3] MATOUŠEK D.:
Práce s mikrokontroléry atmel AT89C2051. Nakladatelství BEN, Praha 2004.
- [4] CHYSKÝ J., NOVÁK J., & NOVÁK L.:
Elektronické aplikace ve strojírenství: Mikroprocesory. Vydavatelství ČVUT, Praha 1998.
- [5] PUNČOCHÁŘ J.:
Operační zesilovače v elektronice. BEN – technická literatura, Praha 1999.
- [6] RAZÍM M., & HORÁČEK P.:
Optimální a adaptivní řízení. Vydavatelství ČVUT, Praha 1985.
- [7] SOUČEK P.:
Pohony výrobních zařízení – servomechanismy. Vydavatelství ČVUT, Praha 1997.
- [8] SUCHÁNEK V.:
Silnoproudá elektrotechnika v automatizaci. SNTL, Praha 1982.
- [9] GM Elektronik, s.r.o.:
Součástky pro elektroniku. Katalog, Praha 2003.
- [10] GES Electronics [online]
<<http://www.ges.cz>>.
- [11] Dallas Semiconductor MAXIM [online]
<<http://www.maxim-ic.com>>.
- [12] ATMEL Microcontrollers [online]
<<http://www.atmel.com>>.
- [13] Datasheetarchive [online]
<<http://www.datasheetarchive.com>>.

Seznam použitých zkratek

IRC	Inkrementální rotační čidlo
LED	Light Emitting Diode (Svítivá dioda)
N.C.	Not Connected (Nezapojeno)
NPN	Negativ Positiv Negativ (Typ tranzistoru)
TTL	Transistor Transistor Logic (Tranzistorově vázaná logika, technologie výroby logických obvodů)
CMOS	Complementary Metal Oxid Semiconductor (technologie výroby integrovaných obvodů)
GND	GrouND (Společný zemní vodič, nulové napětí)
AGND	Analog GrouND (Analogový napěťový zemní vodič)
D/A	Digital to Analog to (Převod digitálního signálu na analogový)
ISP	In System Programmable (Programování jednočipového procesoru)
RAM	Random Access Memmory (Paměť dat mikroprocesoru)
DC	Direct Current (Stejnoseměrný proud)
I ² C	Inter-integrated Circuit Bus (Sériová sběrnice pro propojení integrovaných obvodů)
I/O	Input / Output (Vstupně – výstupní)
PC	Personal Computer (Osobní počítač)
WWW	World Wide Web (Služba umožňující přenos hypertextových souborů)
http	Hyper Text Transfer Protocol (Protokol pro přenos hypertextových souborů)
PID	Proporcionálně Integračně Derivační regulátor (Typ regulátoru)
OZ	(Operační zesilovač)

A.1.2 Charakteristiky ss. motoru

účinnost η [%]=f(M [mN.m]), výkon P [W]=f(M [mN.m]), proud I [A]=f(M [mN.m]),
otáčky n [1/min]=f(M [mN.m])



Obr. A.3. Charakteristiky stejnosměrného servomotoru

A.2 Optoelektronický generátor impulsů: MOZ30

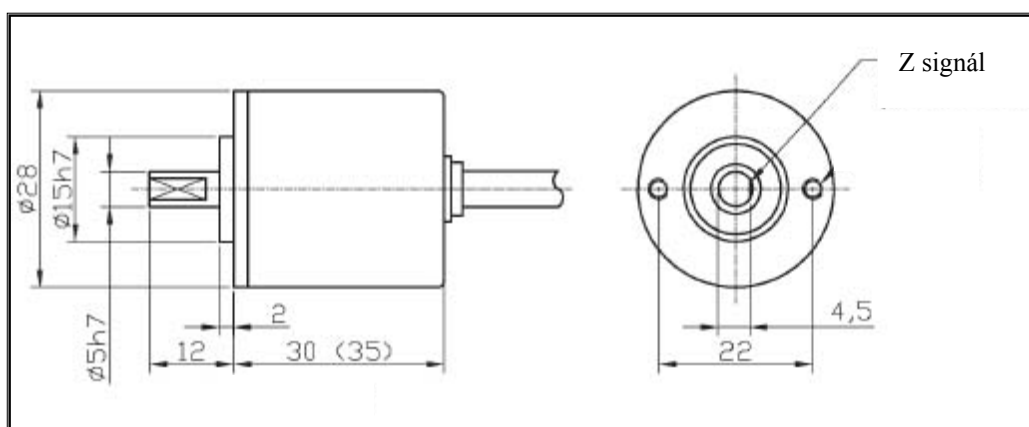


Výrobce: MEGATRON Elektronik
www.megatron.de

Parametry:

- pouzdro o průměru 30 mm
- 2 kanály a referenční impuls
- provozní napětí 5 až 24 V
- přímo volitelný elektronický výstup
- rozměry: Ø28×35 mm

Obr. A.4. IRC snímač MOZ30



Obr. A.5. Rozměrové parametry snímače MOZ30

A.2.1 Elektronické data snímače

Tab. A.1. Elektronické data snímače MOZ30

Tab. 11: Elektrické data šmídacího MOSFETu				
Výstupní el. hodnoty	NPN	K	HK	N
Napájecí napětí [V]	4,5 - 5,5/10,8 - 13,2		21,6-26,4	4,75-5,25
Jmenovité napětí [V]	5/12		24	5
Zaopatřovací proud [mA]	< 80	< 60		< 150
Horní hodnota [V]	> U -1	-		> 2,5
Dolní hodnota [V]	< 0,5			
max. výstupní proud [mA]	< 20			
max. výstupní napětí [V]	-	< 50		-
Mezní frekvence [kHz]	150			

A.2.2 Mechanické, ostatní data snímače MOZ30

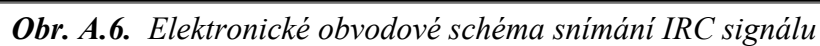
Tab. A.2. *Ostatní data snímače MOZ30*

Druh krytí	IP50
Pracovní teplota	-10 °C..+70 °C
Úložní teplota	-30 °C..+80 °C
Max. nárazové zatížení	490 m/s/11 ms X,Y,Z 3x
Odolnost proti vibraci	10 .. 50 Hz/1,5 mm 2h

Tab. A.3. *Mechanické data snímače MOZ30*

Materiál pouzdra	Hliník
Rozlišení [Imp/U]	36 - 1500 viz vlastnosti
Max. otáčky [1/min]	6000
Životnost ložiska [U]	1,0 E + 8
Axiální zatížení [N]	4,9
Radiální zatížení [N]	9,8
Hmotnost [kg]	0,06

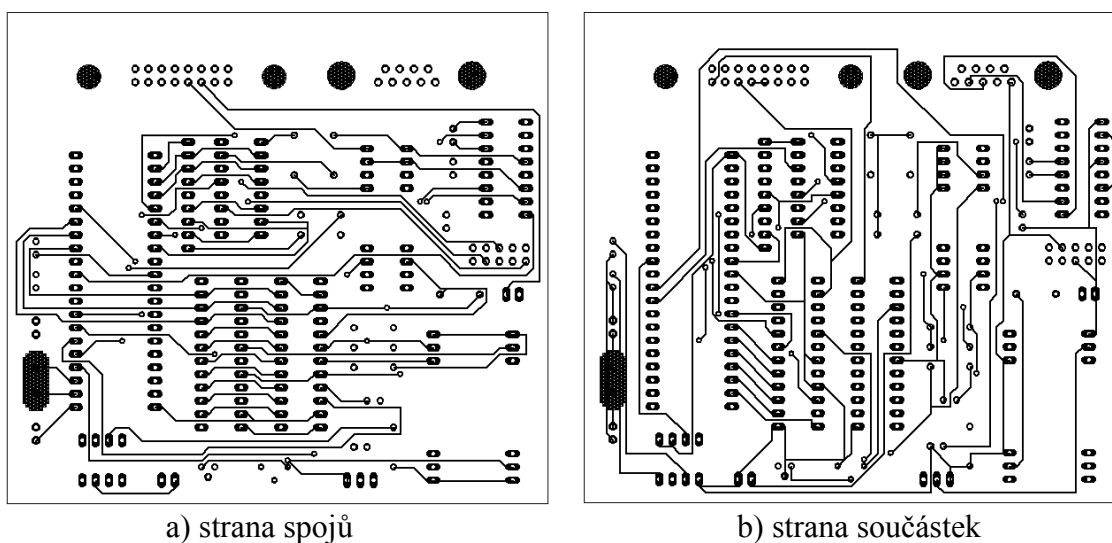
A.3.1 Elektronické schéma snímání IRC signálu



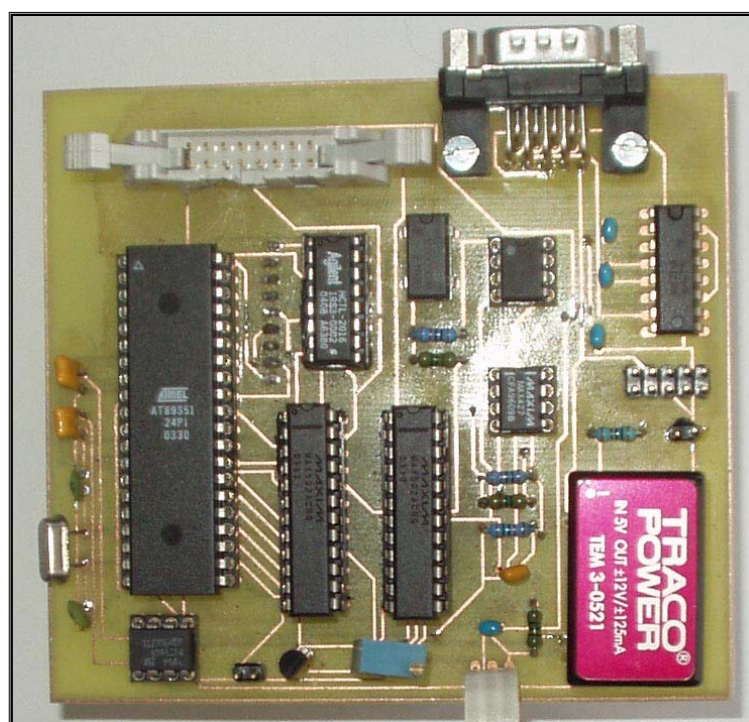
A.3.1.1 Seznam použitých elektronických součástek

<i>Označení</i>	<i>Název-hodnota</i>	<i>Popis</i>
IO1	AT89S51	Mikroprocesor
IO2, IO5	MAX502CNG	D/A převodník
IO3	MAX232	Převodník úrovní RS232 / TTL
IO4	MAX427	Operační zesilovač
IO6	24C02AP	Paměť EPROM
IO7	TL7705	Resetovací obvod
IO8	HCTL2016	Dekodér-čítač IRC
IO9	LM336	Zdroj referenčního napětí
IO10	DC/DC	DC/DC měnič napětí
R1, R2, R3, R4	10 k Ω	Rezistor
R5, R6	10 k Ω	Rezistor
R7	1 k Ω	Rezistor
R8	10 k Ω	Odporový trimr
R9	10 k Ω ×8	Rezistorová síť
C1, C2	33 pF	Kondenzátor
C3, C4	0,1 μ F	Kondenzátor
C5, C6	100 nF	Kondenzátor
C7, C8, C9, C10	10 μ F	Kondenzátor
C12	47 μ F	Kondenzátor
C14, C15	100 μ F	Kondenzátor
OG1	10 MHz	Oscilátor pro HCTL2016
Q1	24 MHz	Krystalový oscilátor pro jednočip
PL1	PSL16	Konektor pro PC
JP1	Konektorový kolík	Vstup IRC signálů
JP2	Konektorový kolík	Přepínač napětí +5 V
JP3	Konektorový kolík	Resetovací mikrotlačítko
JP4	Konektorový kolík	Výstup z DC/DC měniče

A.3.1.2 Deska plošného spoje snímací jednotky

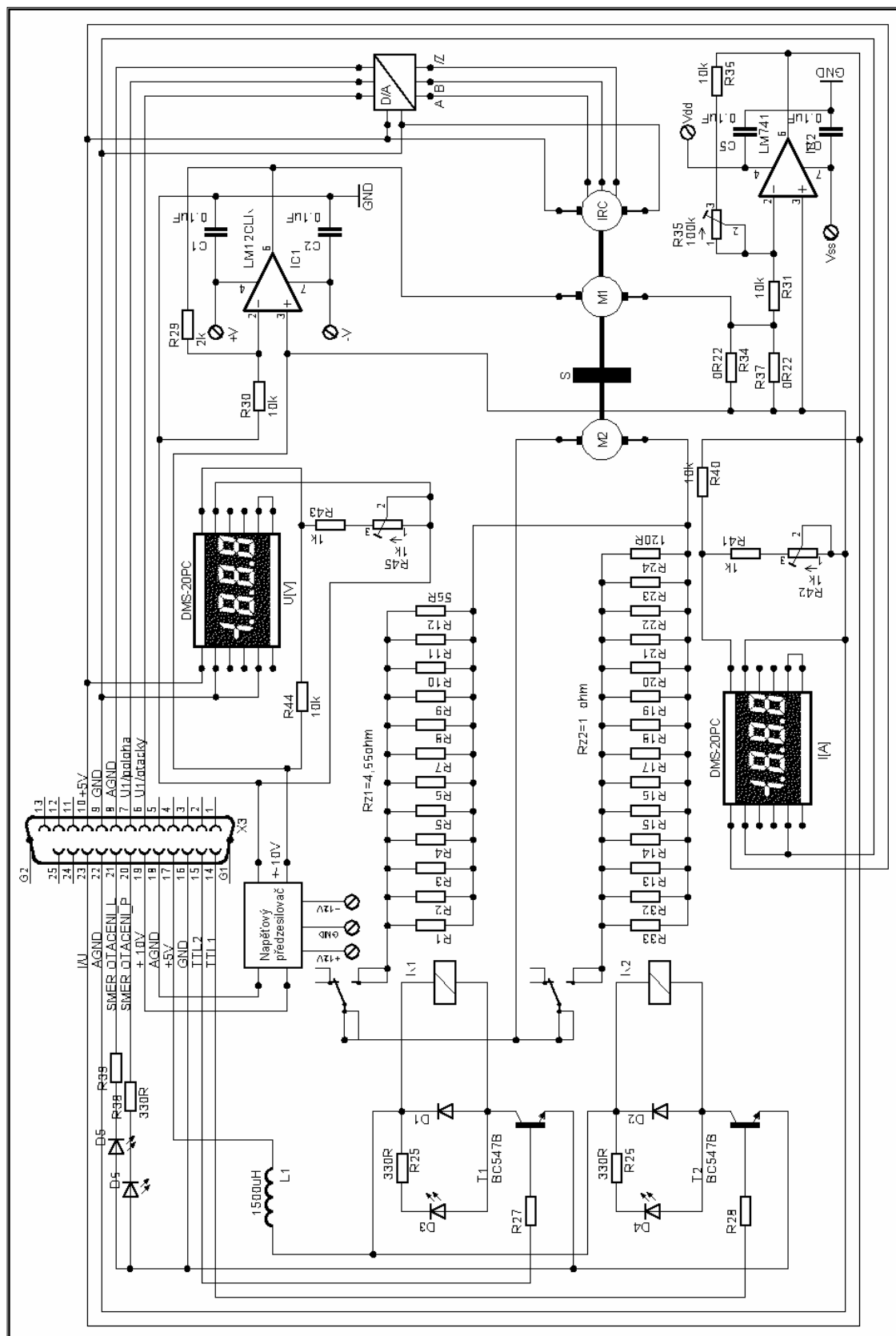


Obr. A.7. Deska plošného spoje snímací jednotky



Obr. A.8. Osazení plošného spoje snímací jednotky

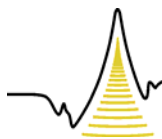
A.3.2 Celkové schéma elektronického zapojení úlohy



Obr. A.9. Celkové obvodové elektronické schéma úlohy

A.3.2.1 Seznam použitých elektronických součástek

<i>Označení</i>	<i>Název-hodnota</i>	<i>Popis</i>
IO1	LM12CLK	Operační výkonový zesilovač
IO2	LM741	Operační zesilovač
R1 – R12	56 Ω	Výkonový rezistor (5 W)
R13 – R24	120 Ω	Výkonový rezistor (5 W)
R25, R26	330 Ω	Rezistor
R27, R28, R29	2 k Ω	Rezistor
R30, R31, R36	10 k Ω	Rezistor
R32, R33	2,2 Ω	Výkonový rezistor (5 W)
R34, R37	0,22 Ω	Rezistor
R35	100 k Ω	O odporový trimr
R38, R39	330 Ω	Rezistor
R40, R44	10 k Ω	Rezistor
R41, R43	1 k Ω	Rezistor
R42, R45	1 k Ω	O odporový trimr
C1, C2, C3, C4	0,1 μ F	Kondenzátor
D1, D2	1N4148	Usměrňovací dioda
D3, D4, D5, D6	LED	LED dioda
T1, T2	BC547B	NPN tranzistor
K1, K2	VS5 5STB	Relé
L1	1500mH	Tlumivka
DMS-20PC	3½ Digit	LED displej
X3	SUB25	Cannon konektor 25 pin.
D/A	Elektronické zapojení vyhodnocující otáčky a polohu ss. motoru	
Napěťový předzesilovací modul	El. Obvod umožňující reverzaci motoru	



A.4 Výpis programů pro optimalizaci PID regulátoru

1) PIDopta.m

```
%PIDopta J=integral[e^2+kappa*(u-unek)^2]
clc; clear all; close all;
global P I D Tsim

R=0.69; %odpor kotvy ss. motoru
Css=0.016; %konstanta motoru
A=[2 1]; %polynom A(s)
B=[1]; %polynom B(s)
C=[1]; %polynom C(s)
Ts=0.05; %perioda vzorkování
Tsim=20; %doba simulace
dT=0.01; %krok simulace

% výchozí zesílení PID regulátoru, N=TV
P0=0.5308; I0=0.2703; D0=0.0403; N=20;
n=length(A);
Kappa=0.5; %váhový koeficient

PoruchaDu=0;
PoruchaD=0,005;
ZadanaW=1;

disp('OPTIMALIZACE PARAMETRU PID-REGULATORU:')
disp('Kriterium:J=integral {e^2 + kappa * (u - unek)^2} :'),Kappa
P=P0; I=I0; D = D0;
sim('pidkra',Tsim);disp('Hodnota kvadr.kriteria pro původní nastavení PID regulátoru:');
krit
x=[P I D];
PIDpoc=x
OPTIONS=optimset('TolFun',1e-10,'MaxFunEvals',300);
x = fminsearch('fpidkra',x,OPTIONS);
disp('Optimalizované parametry PID-Regulátoru:')
PID=x
sim('pidkra',Tsim);disp('Hodnota kritéria pro optimalizované nastavení PID regulátoru:');
krit
sim('pidsima',Tsim); PIDsima;
```

2) fPIDkra.m

```
function f=fPIDkra(x)
global P I D Tsim
P=x(1);
I=x(2);
D=x(3);
sim('pidkra',Tsim);
f=krit;
```



