

**ZÁPADOČESKÁ UNIVERZITA V PLZNI**

**FAKULTA ELEKTROTECHNICKÁ**

KATEDRA ELEKTROMECHANIKY A VÝKONOVÉ ELEKTRONIKY

**BAKALÁŘSKÁ PRÁCE**

Regulace rychlosti stejnosměrného motoru

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Michal KRONEISL  
Osobní číslo: E09B0058P  
Studijní program: B2612 Elektrotechnika a informatika  
Studijní obor: Elektrotechnika a energetika  
Název tématu: Regulace rychlosti stejnosměrného motoru  
Zadávací katedra: Katedra elektromechaniky a výkonové elektroniky

### Z á s a d y p r o v y p r a c o v á n í :


1. Návrh desky plošných spojů pro měnič a řídicí mikroprocesor.
2. Sestavení a oživení úlohy regulace rychlosti stejnosměrného cize buzeného motoru.
3. Zhodnocení dosažených výsledků.

Rozsah grafických prací: podle doporučení vedoucího  
Rozsah pracovní zprávy: 20 - 30 stran  
Forma zpracování bakalářské práce: tištěná/elektronická  
Seznam odborné literatury:

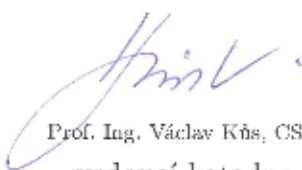
**Student si vhodnou literaturu vyhledá v dostupných pramenech podle doporučení vedoucího práce.**

Vedoucí bakalářské práce: **Ing. Martin Janda, Ph.D.**  
Katedra elektromechaniky a výkonové elektroniky

Datum zadání bakalářské práce: **17. října 2011**  
Termín odevzdání bakalářské práce: **3. června 2012**

  
Doc. Ing. Jirí Hammerbauer, Ph.D.  
děkan



  
Prof. Ing. Václav Kůs, CSc.  
vedoucí katedry

V Plzni dne 17. října 2011

# **Anotace**

Kroneisl, M. Regulace rychlosti stejnosměrného motoru, Západočeská univerzita v Plzni, Fakulta elektrotechnická, Katedra elektromechaniky a výkonové elektroniky, 2012. Vedoucí práce: Ing. Martin Janda Ph.D.

Tato bakalářská práce se zabývá návrhem pulsního měniče pro stejnosměrný cize buzený motor a číslicovým řízením tohoto měniče pomocí mikrokontroléru PIC.

# **Klíčová slova**

regulace rychlosti, číslicová regulace, pulsní měnič, stejnosměrný motor, motor s cizím buzením, mikrokontrolér, PIC

# **Annotation**

Kroneisl, M. DC machine speed control, University of West Bohemia, Faculty of electrical engineering, 2012. Head: Ing. Martin Janda Ph.D.

This bachelor thesis describes the design of DC chopper for separately excited DC motor and the digital control of the chopper by the PIC microcontroller.

# **Key words**

speed control, digital control, DC chopper, DC motor, separately excited DC motor, microcontroller, PIC

## **Prohlášení**

Předkládám tímto k posouzení a obhajobě bakalářskou práci vypracovanou na závěru mého studia na Fakultě elektrotechnické Západočeské univerzity v Plzni. Prohlašuji, že jsem tuto bakalářskou práci vypracoval zcela samostatně, pouze s použitím literatury a zdrojů uvedených v seznamu, který je součástí práce.

V Plzni dne .....

.....

Michal Kroneisl

## **Poděkování**

Tímto bych rád poděkoval Ing. Martinu Jandovi Ph.D. za cenné připomínky a konzultace během tvorby této bakalářské práce.

# Obsah

Seznam použitých zkratk	9
1. Úvod	10
1.1 Cíle a náplň práce	10
2. Stejnosměrný motor	11
2.1 Náhradní schéma a rovnice	11
2.2 Stejnosměrný motor s permanentními magnety	12
3. Pulsní měnič	13
3.1 Zapojení a způsob řízení	13
3.2 Integrovaný obvod L6203	14
4. Mikrokontrolér Microchip PIC18F2520	16
4.1 Obecné informace	16
4.2 Modul TIMER0	16
4.3 Modul TIMER2	17
4.4 Modul A/D převodníku	18
5. Měření proudu a otáček	21
5.1 Měření proudu	21
5.2 Měření otáček	21
6. Zapojení a konstrukce	22
6.1 Blokové schéma	22
6.2 Elektrické schéma	22
6.3 Plošný spoj	25
6.4 Osazovací plán	26
6.5 Seznam součástek	27
7. Regulace stejnosměrného motoru	28
7.1 Schéma regulace	28
7.2 Řídící algoritmus	28
7.3 Proudová smyčka	31
8. Programové řešení	33
9. Závěr	37
Seznam použité literatury	38

# Seznam použitých zkratek

$I_b$	budící proud motoru
$I_d, i_d$	proud kotvy motoru, proud odebíraný z měniče
$J$	moment setrvačnosti
$k_I$	konstanta PS-regulátoru určující rychlost integrace
$k_R$	konstanta regulátoru
$k_{ZI}$	konstanta měření proudu
$k_{Z\omega}$	konstanta měření otáček
$k_V$	„zesílení“ pulsního měniče
$k\Phi$	konstanta stroje
$M$	mechanický moment motoru
$P_{EL}$	elektrický výkon
$\Delta P_J$	Jouleovy ztráty
$P_{MECH}$	mechanický výkon
$R_A$	odpor kotvy motoru
$L$	indukčnost zátěže
$U_d, u_d$	napětí na hlavních svorkách motoru, napětí na zátěži měniče
$U_i, u_i$	indukované napětí
$U_I$	výstupní napětí převodníku proudu na napětí
$U_\omega$	výstupní napětí tachodynamu
$u_i$	napětí odpovídající proudu zátěže $I_d$
$T$	perioda výpočtu regulátoru
$T_{ACQ}$	doba získávání vzorku (A/D převodník)
$T_{AD}$	doba převodu jednoho bitu (A/D převodník)
$T_{OSC}$	perioda oscilátoru mikrokontroléru
$T_{PWM}$	perioda pulsně šířkové modulace
$x$	výstup regulátoru
$\varepsilon$	regulační odchylka
$\Phi$	magnetický (indukční) tok
$\tau_e$	časová konstanta vinutí kotvy
$\tau_R$	časová konstanta PI-regulátoru
$\tau_V$	časová konstanta pulsního měniče
$\omega$	úhlová rychlost v rad/s
$\omega_f$	úhlová frekvence řezu





# 1. Úvod

## 1.1 Cíle a náplň práce

Cílem této bakalářské práce je navrhnout a realizovat pulsní měnič pro stejnosměrný motor s mikroprocesorovou regulací otáček. Práce shrnuje použitou teorii z oblasti elektrických strojů, výkonové elektroniky a regulační techniky. Je zde popsána funkce a způsob zapojení či nastavení použitých součástek a integrovaných obvodů, zejména pak konfigurace mikrokontroléru PIC a využití jeho jednotlivých modulů. Následuje návrh elektrického zapojení a desky plošných spojů pro pulsní měnič a řídicí a komunikační elektroniku. Závěr práce se zabývá číslicovou realizací PS-regulátoru a nakonec je uveden příklad programu pro řízení otáček motoru.

## 2. Stejnosměrný motor

### 2.1 Náhradní schéma a rovnice

Náhradní schéma kotvy stejnosměrného motoru je na obr. 1. Podle schématu je pro obvod kotvy možné psát rovnice:

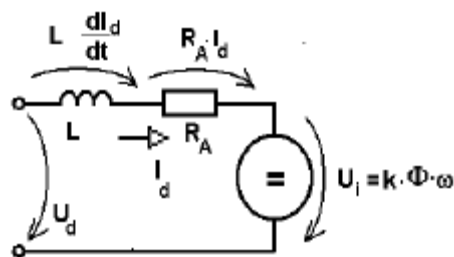
$$u_d = u_L + u_R + U_i \quad (2.1)$$

$$u_d = L \frac{di_d}{dt} + R_A i_d + U_i \quad (2.2)$$

kde  $L$  a  $R_A$  jsou indukčnost a odpor kotvy a  $U_i$  je indukované napětí, pro které platí:

$$U_i = k\Phi\omega \quad (2.3)$$

kde  $\Phi$  je magnetický tok vybuzený statorem a  $\omega$  jsou otáčky rotoru v rad/s. Indukčnost kotvy  $L$  se uplatní pouze při přechodových dějích a vyhlazuje proud kotvou  $I_d$ , proto je žádoucí zejména při napájení motoru z měniče a v některých případech se její hodnota záměrně zvětšuje o další indukčnost zapojenou sériově k motoru. Naproti tomu odpor  $R_A$  „změkčuje“ momentovou charakteristiku a způsobuje Jouleovy ztráty (proto je snaha ho minimalizovat).



obr.2.1. Náhradní schéma kotvy ss motoru [1]

Zdroj indukovaného napětí  $U_i$  představuje samotnou elektromechanickou přeměnu energie, přičemž mezi elektrickým a mechanickým výkonem platí rovnost:

$$P_{MECH} = U_i I_d = k\Phi\omega I_d = M\omega \quad (2.4)$$

Úpravami lze pak získat rovnice pro otáčky a moment motoru:

$$\omega = \frac{U_i}{k\Phi} \quad (2.5)$$

$$M = k\Phi I_d \quad (2.6)$$

Pro vyjádření otáček v závislosti na svorkovém napětí  $U_d$  je třeba respektovat úbytek napětí na odporu  $R_A$  a vztah (2.5) upravit:

$$\omega = \frac{U_i}{k\Phi} = \frac{U_d - R_A I_d}{k\Phi} = \frac{U_d}{k\Phi} - \frac{R_A I_d}{k\Phi} = \frac{U_d}{k\Phi} - \frac{R_A M}{(k\Phi)^2} \quad (2.7)$$

Ze vztahu (2.7) je vidět, že pokles otáček (poslední zlomek) je přímo úměrný odporu kotvy a mechanického momentu stroje. Čím je tedy odpor  $R_A$  větší, tím větší je pokles otáček při zatížení (momentová charakteristika je „měkčí“).

Pokud mechanické ztráty zahrneme do mechanického výkonu, lze energetickou bilanci

stejnosměrného stroje vyjádřit rovnicemi:

$$P_{EL} = \Delta P_J + P_{MECH} \quad (2.8)$$

$$U_d I_d = R_A I_d^2 + U_i I_d \quad (2.9)$$

přičemž při  $P_{EL}$  i  $P_{MECH} > 0$  pracuje stroj jako motor a při  $P_{EL}$  i  $P_{MECH} < 0$  pracuje stroj jako generátor.

## 2.2 Stejnosměrný motor s permanentními magnety

Magnetický tok  $\Phi$  lze vytvořit buďto statorovým (budícím) vinutím nebo permanentními magnety umístěnými na statoru. V prvním případě rozlišujeme podle zapojení budícího vinutí vzhledem k vinutí kotvy stejnosměrné stroje sériové, paralelní, kompaundní (kombinace obou předchozích) a s cizím buzením, přičemž toto rozdělení má zásadní vliv na vlastnosti stroje (např. momentovou charakteristiku) a určuje oblast použití (např. elektrická trakce, spotřebiče s nutností běhu naprázdno, atd.). Stejnosměrné stroje s permanentními magnety mají obdobné vlastnosti jako stroje s cizím buzením a konstantním budícím proudem ( $I_b \sim k\Phi = \text{konst.}$ ).

Pokud  $k\Phi = \text{konst.}$ , je ze vztahů (2.6) a (2.7) zřejmé, že moment stroje lze přímo řídit proudem  $I_d$ , zatímco otáčky stroje naprázdno budou přibližně odpovídat napětí  $U_d$  (resp. jeho střední hodnotě).

### 3. Pulsní měnič

#### 3.1 Zapojení a způsob řízení

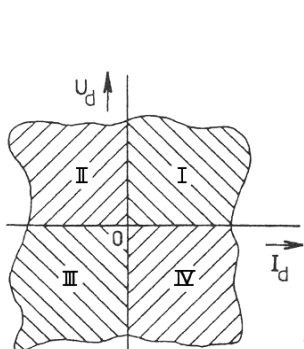
Pokud je potřeba plynule regulovat otáčky či moment motoru, je nutné nějakým způsobem řídit proud kotvy a napětí. Regulace může být buď ztrátová (např. odpor zapojený v sérii s kotvou) nebo „bezztrátová“ s použitím pulsního měniče složeného z výkonových polovodičových prvků provozovaných ve spínacím režimu. Podle toho, zda je měnič schopen měnit polaritu výstupního napětí a proudu rozlišujeme pulsní měniče na jednkvadrantové, dvoukvadrantové a čtyřkvadrantové (viz obr. 3.1).

Příklad jednoduchého jednkvadrantového pulsního měniče je na obr. 3.2. Pokud výkonový prvek V řídíme pulsně šířkovou modulací (PWM), dostaneme na výstupu obdélkové napětí, pro jehož střední hodnotu (při ideálních spínacích prvcích) platí:

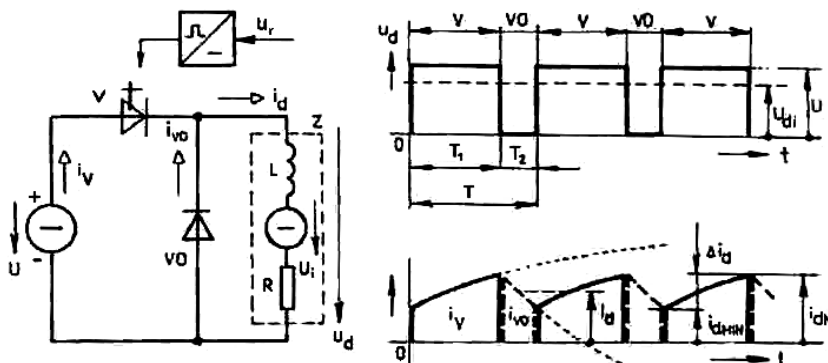
$$U_d = \frac{T_1}{T} U = zU \tag{3.1}$$

kde  $z$  je poměrná doba sepnutí prvku V. Proud je vyhlazen indukčností L a jeho střední hodnotu získáme podle vztahu:

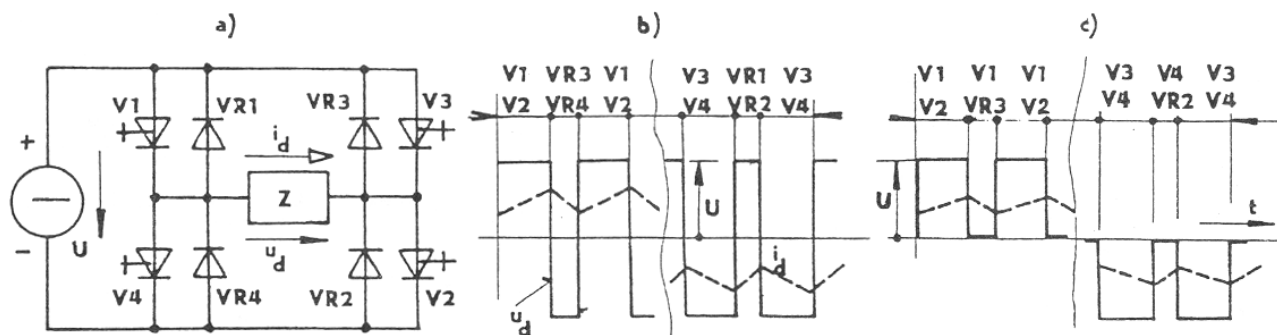
$$I_d = \frac{U_d - U_i}{R} \tag{3.2}$$



obr.3.1. Čtyři kvadranty podle polarity  $U_d$  a  $I_d$  [2]



obr.3.2. Jednkvadrantový snižovací pulsní měnič [2]



obr.3.3. Schéma H-můstku a průběhy  $u_d$  a  $i_d$  při různých způsobech řízení [2]

Pokud je potřeba měnit polaritu napětí i proudu, lze použít čtyřkvadrantový pulsní měnič, tzv. H-můstek (obr. 3.3).

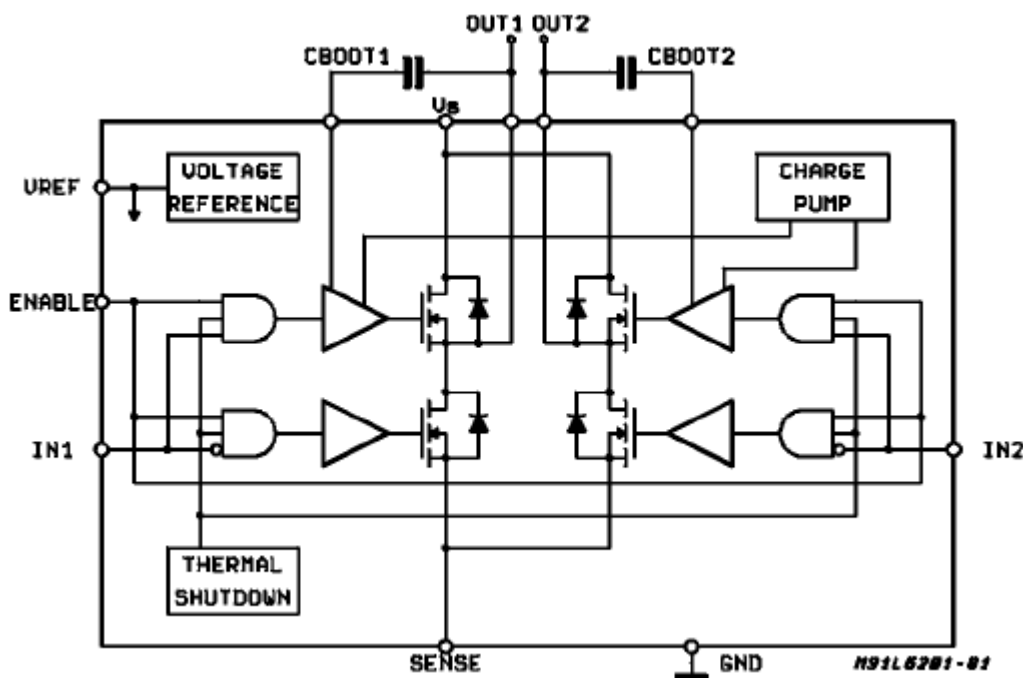
H-můstek lze řídit dvěma způsoby. První způsob spočívá současném spínání vždy dvou protilehlých prvků (obr.3.3b). Střední hodnota napětí na zátěži pak je:

$$U_d = U(2z-1) \quad (3.3)$$

Při druhém způsobu řízení určuje sepnutí jednoho z prvků V1 nebo V4 polaritu napětí, zatímco prvek V2 nebo V3 je řízen pulsně šířkovou modulací (PWM), která určuje velikost napětí. Pokud se nemění stav prvků V1 a V4, funguje měnič jako dvoukvadrantový s reverzací proudu. Tento způsob řízení se oproti prvnímu vyznačuje rovněž menším zvlněním proudu.

### 3.2 Integrovaný obvod L6203

Integrovaný obvod L6203 od firmy SGS-Thomson obsahuje H-můstek složený z tranzistorů DMOS, budící obvody tranzistorů a tepelnou ochranu. Vnitřní blokové schéma IO je na obr. 3.4. Můstek je řízen třemi logickými vstupy pracujícími s úrovněmi TTL. Vstupy IN1 a IN2 určují výstupní napětí jednotlivých půlmůsteků (na svorkách OUT1 a OUT2). Vstup ENABLE pak (de)aktivuje celý H-můstek. Pokud přivedeme na vstup ENABLE nízkou úroveň (L), jsou všechny tranzistory rozepnuty a oba výstupy ve stavu „vysoké impedance“ (viz tab. 3.1). Pro správnou činnost můstku je třeba připojit vnější kapacity CBOOT1 a CBOOT2, které slouží nábojové pumpě (CHARGE PUMP) k vytvoření potřebného řídicího napětí pro horní dva tranzistory. H-můstek má rovněž zvlášť vyvedenou zem z H-můstku (svorka SENSE), kterou lze použít k připojení rezistoru pro měření proudu. Pokud není rezistor zapojen, propojí se se zemí (GND).



obr.3.4. Vnitřní blokové schéma integrovaného obvodu L6203 [W1]

tab.3.1. Pravdivostní tabulka obvodu L6203

řídící vstupy (TTL)			výstupní napětí	
ENABLE	IN1	IN2	OUT1	OUT2
H	L	L	0	0
H	L	H	0	U
H	H	L	U	0
H	H	H	U	U
L	x	x	Z	Z

pozn.: x – nezáleží na hodnotě, Z – stav „vysoké impedance“

Některé základní parametry použitého IO L6203 v pouzdře Multiwatt11 [W1]:

Napájecí napětí:	12-48V
Trvalý proud (RMS):	4A
Maximální spínací frekvence:	100 kHz
Mezní ztrátový výkon:	20W

## 4. Mikrokontrolér Microchip PIC18F2520

### 4.1 Obecné informace

PIC18F2520 od firmy Microchip je osmibitový jednočipový mikrokontrolér s RISC jádrem a Harvardskou architekturou. Program je uložen v 32KB paměti typu flash, datová paměť je statická RAM s kapacitou 32 KB, což vystačí na 16384 jednoslovných instrukcí. Kromě toho obsahuje 256 B EEPROM, 10-bitový A/D převodník, 2 Capture/Compare/PWM moduly, sériové rozhraní USART a 4 čítače. Maximální frekvence při použití krystalového oscilátoru s PLL je 40 Mhz. Mikrokontrolér má 3 8-bitové vstupní/výstupní brány.

### 4.2 Modul TIMER0

TIMER0 je čítač použitý pro měření času. Při jeho přetečení dojde k přerušení, které následně vyvolá provedení výpočetního algoritmu regulátoru. Frekvence, se kterou dochází k přetečení čítače, je stejná, jako frekvence PWM. K nastavení modulu TIMER0 slouží registr T0CON (tab. 4.1 a 4.2), k nastavení přerušení pak registr INTCON (tab. 4.3 a 4.4).

tab.4.1. Registr T0CON [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	TMROON	T08BIT	T0CS	T0SE	PSA	T0PS2	T0PS1	T0PS0

tab.4.2. Nastavení registru T0CON [W2]

bit	možnosti	nastavení
TMROON Zapnutí čítače	0 = vypnuto 1 = zapnuto	1
T08BIT 8 nebo 16-bitový režim	0 = 16-bitový režim 1 = 8-bitový režim	1
T0CS Nastavení vstupu čítače	0 = vstupem čítače je vnitřní oscilátor 1 = externí vstup čítače	0
T0SE Nastavení hrany vstup. signálu	při použití vnitřního oscilátoru nemá význam	0
PSA Zapnutí předděličky	0 = zapnuta předdělička 1 = vypnuta předdělička	0
T0PS2 : T0PS0 Dělicí poměr předděličky	111 = 1:256 ... 001 = 1:4 000 = 1:2	001

tab.4.3. Registr INTCON [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	GIE/GIEH	PEIE/GIEL	TMROIE	INT0IE	RBIE	TMROIF	INT0IF	RBIF

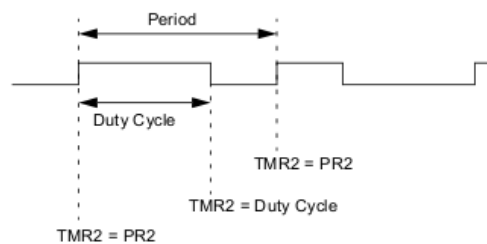


tab.4.4. Nastavení registru INTCON [W2]

bit	možnosti	nastavení
GIE/GIEH Maskování všech přerušení	0 = všechna přerušení zakázána 1 = všechna nemaskovaná přerušení povolena	1
PEIE/GIEL Povolení přerušení od periférií	0 = zakázány 1 = povoleny	1
TMR0IE Povolení přerušení od TIMER0	0 = zakázáno 1 = povoleno	1
TMR0IF Příznak přerušení od TIMER0	0 = nedošlo k přerušení 1 = došlo k přerušení (přetečení čítače TIMER0)	0

### 4.3 Modul TIMER2

TIMER2 je 8-bitový čítač použitý spolu s capture/compare/pwm modulem CCP2 pro generování pulsně šířkové modulace (PWM). Stav čítače je uložen v registru TMR2. Tento registr je porovnáván se stavem registrů PR2 (určuje periodu PWM) a CCPR2L (určuje střidu PWM). Pokud čítač dosáhne hodnoty uložené v registru CCPR2L, je výstup PWM vynulován, v případě, že čítač dosáhne hodnoty uložené v registru PR2, je výstup naopak nastaven na logickou 1 (viz obr. 4.1). V případě, že CCPR2L = 0, výstup se nenastavuje a zůstává stále v logické 0.



obr.4.1. PWM výstup [W2]

Střida (poměrná doba sepnutí výstupu) PWM se pak určí podle vztahu:

$$z = \frac{CCPR2L}{PR2} \quad (4.1)$$

Periodu PWM lze určit jako:

$$T_{PWM} = 4T_{OSC}(\text{prescale})(PR2 + 1) \quad (4.2)$$

kde  $4T_{OSC}$  je doba instrukčního cyklu a „prescale“ je dělicí poměr předděličky čítače.

Př.:

- pro  $4T_{OSC} = 2.10^{-7}$  s (pro 20 MHz oscilátor)  
 $PR2 = 255$  (maximální perioda, „nejjemnější“ nastavení střidy)  
 $\text{prescale} = 4$

vychází perioda  $T_{PWM} = 2,048.10^{-4}$  s a z toho frekvence  $f_{PWM} = 4882,8125$  Hz.

Další konfigurační registry jsou T2CON (nastavení čítače TIMER2), CCP2CON (nastavení capture/compare/pwm modulu CCP2). Pro správnou funkci PWM je rovněž třeba výstup modulu CCP2 přiřadit bráně B (nastaví se v registru CONFIG3H při programování procesoru) a nastavit vývod RB3 jako výstup (3. bit registru TRISB = 0).

tab.4.5. Registr T2CON [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	-	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0

tab.4.6. Nastavení registru T2CON [W2]

bit	možnosti	nastavení
TMR2ON Zapnutí čítače TIMER2	0 = čítač vypnut 1 = čítač zapnut	1
T2CKPS1 : T2CKPS0 Dělicí poměr předděličky čítače TIMER2	00 = 1:1 01 = 1:4 1x = 1:16	01

tab.4.7. Registr CCP2CON [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	-	-	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0

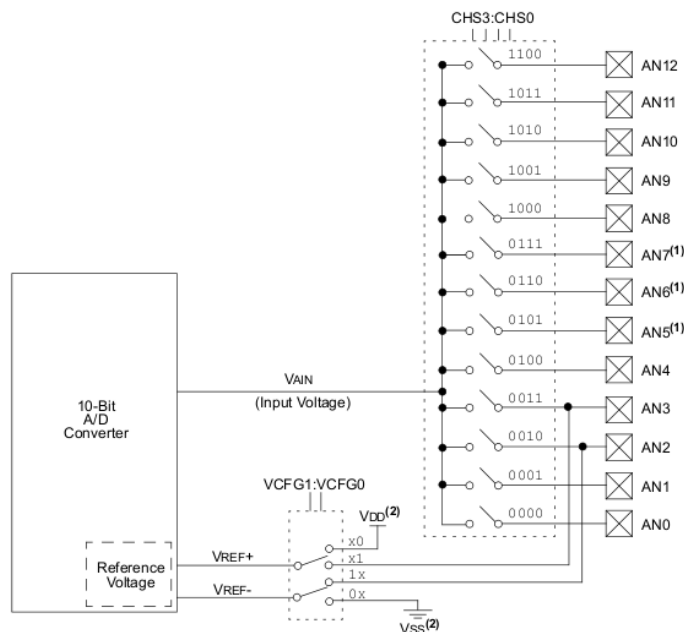
tab.4.8. Nastavení registru CCP2CON [W2]

bit	možnosti	nastavení
DC2B1 : DC2B0 - Nejnižší 2 bity střídy PWM (PWM je ve skutečnosti 10-bitová), jsou porovnávány s nejvyššími 2 bity předděličky čítače. Kvůli výpočtové náročnosti nepoužity a PWM funguje jako 8-bitová.		00
CCP2M3 : CCP2M0 režim capture/compare/pwm modulu	0000 = capture/compare/pwm vypnuto 0001 až 1011 = různé režimy capture/compare 11xx = PWM režim	1100

## 4.4 Modul A/D převodníku

Integrovaný A/D převodník je použit pro měření proudu a otáček převedených na napětí (viz. kapitola Měření proudu a otáček). Modul A/D převodníku je v mikrokontroléru pouze jeden, proto je třeba při měření více signálů je na analogové straně multiplexovat (viz obr. 4.2). Dohromady je k dispozici 10 analogových kanálů (AN5 až AN7 nejsou u modelu 18F2520 použity). Kromě toho lze také zvolit referenční napětí, které může být odvozeno buď z napájení procesoru ( $V_{SS}$ ,  $V_{DD}$ ) nebo z externího zdroje (AN2 a AN3).

Při měření proudu je převáděn vstup AN1 a pro zvětšení dynamického rozsahu je zvolen vnější zdroj referenčního napětí ( $V_{REF+} = 4V$ ,  $V_{REF-} = 1V$ ). Při měření otáček je převáděn vstup AN0 a jako referenční napětí slouží napájení ( $V_{DD} - V_{SS} = 5V$ ). Pro měření obou veličin v reálném čase se oba vstupy i referenční napětí střídavě přepínají.



obr.4.2. Vstupy A/D převodníku

Nastavení modulu A/D převodníku je provedeno pomocí registrů ADCON0, ADCON1 a ADCON2. Pro registry ADCON0 a ADCON1 je v programu použito dvoje nastavení podle toho, zda se převádí proud nebo otáčky. Nastavení registrů viz tab. 4.9 až 4.12.

tab.4.9. Registr ADCON0 [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	-	-	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON

tab.4.10. Nastavení registru ADCON0 [W2]

bit	možnosti	nastavení	
		proud	otáčky
CHS3 : CHS0 Výběr převáděného kanálu	0000 = kanál AN0 0001 = kanál AN1 ... 1100 = kanál AN12	0000	0001
GO/DONE Stav A/D převodníku	0 = A/D převod neprobíhá (byl dokončen) 1 = A/D převod právě probíhá	1	
ADON Zapnutí A/D převodníku	0 = A/D převodník vypnut 1 = A/D převodník zapnut	1	

tab.4.11. Registr ADCON1 [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	-	-	VCFG1	VCFG0	PCFG3	PCFG2	PCFG1	PCFG0

tab.4.12. Nastavení registru ADCON1 [W2]

bit	možnosti	nastavení	
		proud	otáčky
VCFG1 Nastavení záporné reference	0: $V_{REF-} = V_{SS}$ 1: $V_{REF-} = AN2$	1	0
VCFG0 Nastavení kladné reference	0: $V_{REF+} = V_{DD}$ 1: $V_{REF+} = AN3$	1	0
PCFG3 : PCFG0 Nastavení vývodů jako analogových nebo digitálních	0000 = AN0 až AN12 jsou analogové vstupy ... 1011 = AN0 až AN3 jsou analogové vstupy ... 1111 = AN0 až AN12 jsou digitální vstupy/výstupy	1011	

Časování průběhu A/D převodu je nastaveno registrem ADCON2 (tab. 4.13 a 4.14). Nastavení je důležité jak pro získání správného výsledku, tak i z hlediska požadované rychlosti převodu (a tím dané vzorkovací frekvence). Průběh celého A/D převodu lze rozdělit na 2 části: dobu získávání vzorku  $T_{ACQ}$ , kdy je z analogového vstupu nabíjen tzv. vzorkovací kondenzátor a dobu samotného převodu.

Použitý A/D převodník je s tzv. postupnou aproximací, to znamená, že převádí hodnotu postupně po jednotlivých bitech od nejvyššího (MSB) po nejnižší (LSB). Doba převodu jednoho bitu  $T_{AD}$  je odvozena od periody oscilátoru nastavitelnou předděličkou (reg. ADCON2, bity ADCS2 : ADCS0). Převedení všech 10 bitů vyžaduje celkem 11 period  $T_{AD}$ . Doba získávání vzorku  $T_{ACQ}$  (nabíjení kondenzátoru) je nastavitelná v násobcích  $T_{AD}$  (reg. ADCON2, bity ACQT2 : ACQT0).

Rovnice pro určení optimálního načasování A/D převodu jsou uvedeny ve [W2]. Zvolené nastavení je  $T_{AD} = 16 T_{OSC}$  a  $T_{ACQ} = 4 T_{AD}$ . Celková doba převodu i s vybitím kondenzátoru na konci je  $(4+11+1) T_{AD} = 256 T_{OSC} = 12,8 \mu s$  (pro 20 MHz krystal).

tab.4.13. Registr ADCON2 [W2]

pořadí bitu	7	6	5	4	3	2	1	0
název bitu	ADFM	-	ACQT2	ACQT1	ACQT0	ADCS2	ADCS1	ADCS0

tab.4.14. Nastavení registru ADCON2 [W2]

bit	možnosti	nastavení
ADFM Zarovnání výsledku převodu	0 = zarovnání doleva 1 = zarovnání doprava	1
ACQT2 : ACQT0 Doba získávání vzorku $T_{ACQ}$	111 = $20 T_{AD}$ ... 010 = $4 T_{AD}$ ... 001 = $2 T_{AD}$	010
ADCS2 : ADCS0 Zdroj hodinových impulsů pro A/D převodník	110 = $F_{OSC} / 64$ 101 = $F_{OSC} / 16$ ... 000 = $F_{OSC} / 2$	101

Po dokončení A/D převodu je jeho výsledek uložen do registrů ADRESH a ADRESL. Protože je A/D převodník 10-bitový, lze zvolit zarovnání výsledku doleva nebo doprava (bit ADFM v registru ADCON2).

## 5. Měření proudu a otáček

### 5.1 Měření proudu

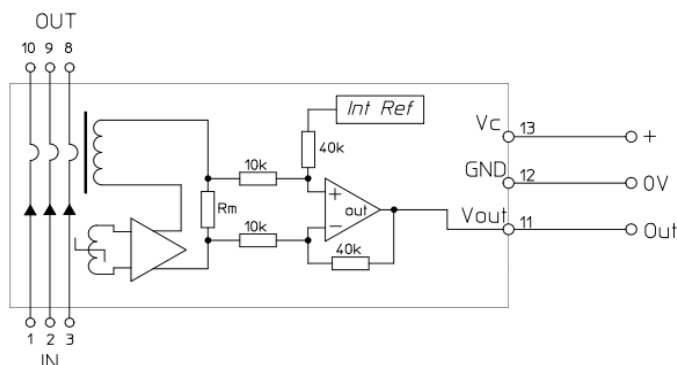
Pro měření proudu je použito čidlo typu CAS 15-NP od firmy LEM. Jedná se o převodník proudu na napětí, který pracuje na principu měření magnetického pole vyvolaného proudem pomocí Hallovy sondy. Díky tomu je schopno měřit střídavé, stejnosměrné i pulsní průběhy proudu se šířkou pásma od 0 do 200 kHz ( $\pm 1$  dB). Měřený proud a výstupní napětí jsou galvanicky odděleny, měřicí elektronika je napájena napětím 5 V.

Vnitřní schéma je na obr. 5.1. Převodník má celkem 3 primární vinutí (svorky 1-3 a 8-10), podle jejichž zapojení lze nastavovat měřicí rozsah. Jmenovitý rozsah u typu 15-NP je  $\pm 15$  A, při zapojení všech tří vinutí do série pak lze měřit proud v rozsahu  $\pm 5$  A. Magnetický tok vyvolaný primárními vinutím je kompenzován vinutím sekundárním tak, aby výsledný magnetický tok bylo roven nule. Proud potřebný ke kompenzaci je měřicím odporem  $R_m$  převeden na napětí a zesílen.

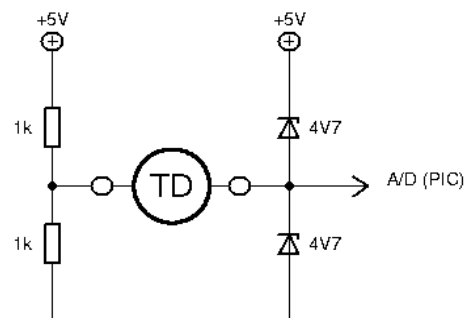
Výstupní napětí převodníku  $u_I$  (na obrázku značeno  $V_{OUT}$ ) je pro typ 15-NP při sériovém zapojení primárních vinutí dáno vztahem:

$$U_I = 2,5 + 0,125 I_d \text{ [V ; A]}$$

kde  $I_d$  je měřený proud zátěže. Při rozsahu  $I_d \pm 5$  A se tedy napětí mění v rozmezí od 1,875 do 3,125 V. Podle toho je upraveno referenční napětí na výstup připojeného A/D převodníku (viz kapitola 4.4).



obr.5.1. Vnitřní schéma převodníku LEM řady CAS [W3]



obr.5.2. Připojení tachodynamu

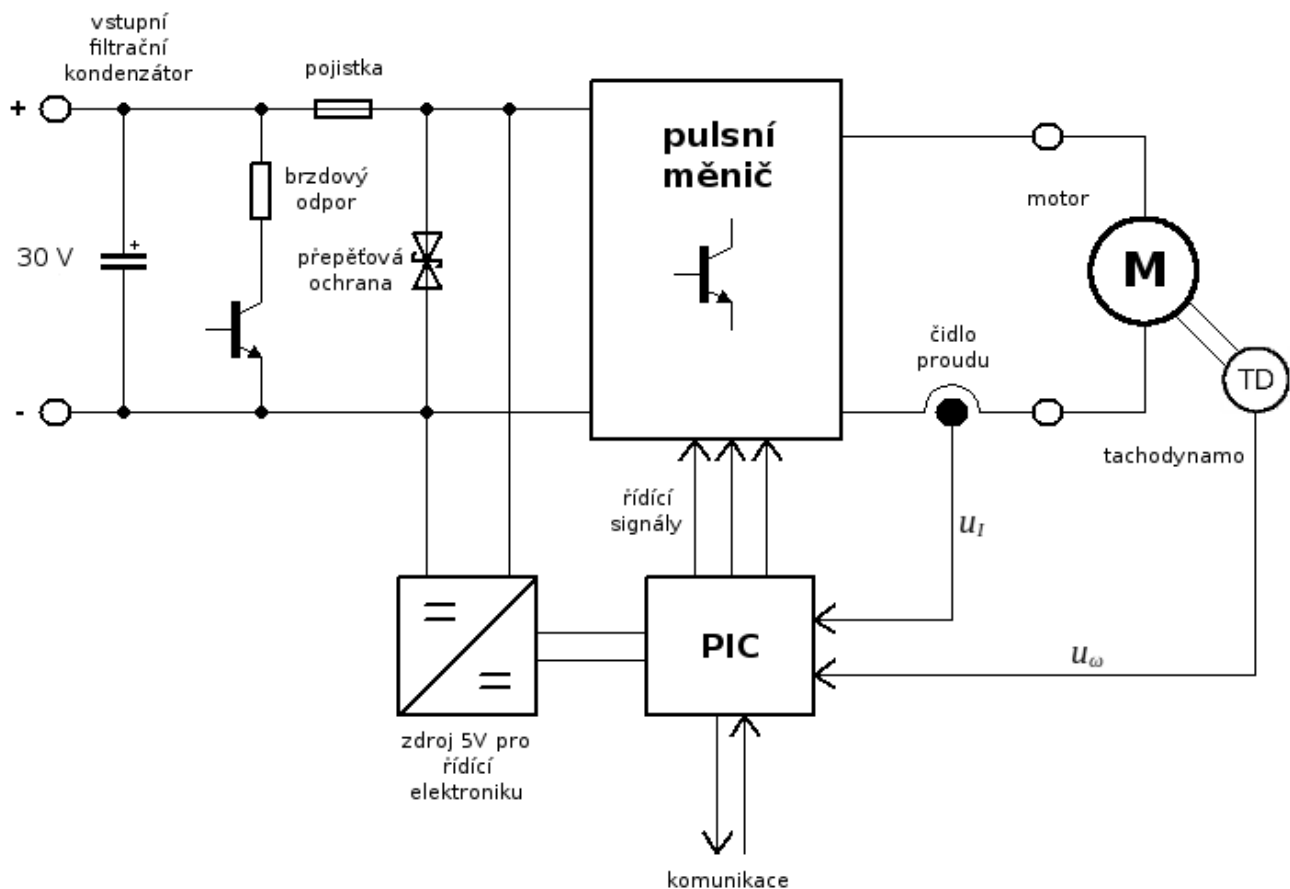
### 5.2 Měření otáček

Otáčky jsou měřeny tachodynamem typu K5A9-00 od firmy ATAS Náchod, které je k motoru připojeno přes pružnou spojku. Měřením bylo zjištěno, že výstupní napětí tachodynamu je při napájení motoru napětím 30 V (předpokládané napájecí napětí měniče, kterému odpovídá přibližně 2850 ot./min.) a běhu naprázdno 2 V. Aby bylo možné měřit obě polarizace výstupního napětí, je tachodynamo připojeno k A/D převodníku mikrokontroléru podle obr. 5.2.

## 6. Zapojení a konstrukce

### 6.1 Blokové schéma

Blokové schéma celého měniče je na obr. 6.1. Celé zapojení je možné napájet ze stejnosměrného zdroje 30 V, doporučené nastavení proudového omezovače zdroje je 2,5 až 3 A. Pokud napájecí napětí překročí cca 43 V, je aktivován brzdový rezistor o hodnotě 10  $\Omega$ . Pro jednoduchost zapojení pracuje „spínací“ tranzistor brzdového rezistoru v lineárním režimu, což vzhledem k výkonu a momentu setrvačnosti motoru postačuje. Rychlá trubičková pojistka má hodnotu 2,5 A. Pro případ jejího přepálení je na vstupu měniče přepětová ochrana v podobě bipolárního transilu s průrazným napětím 47 V.

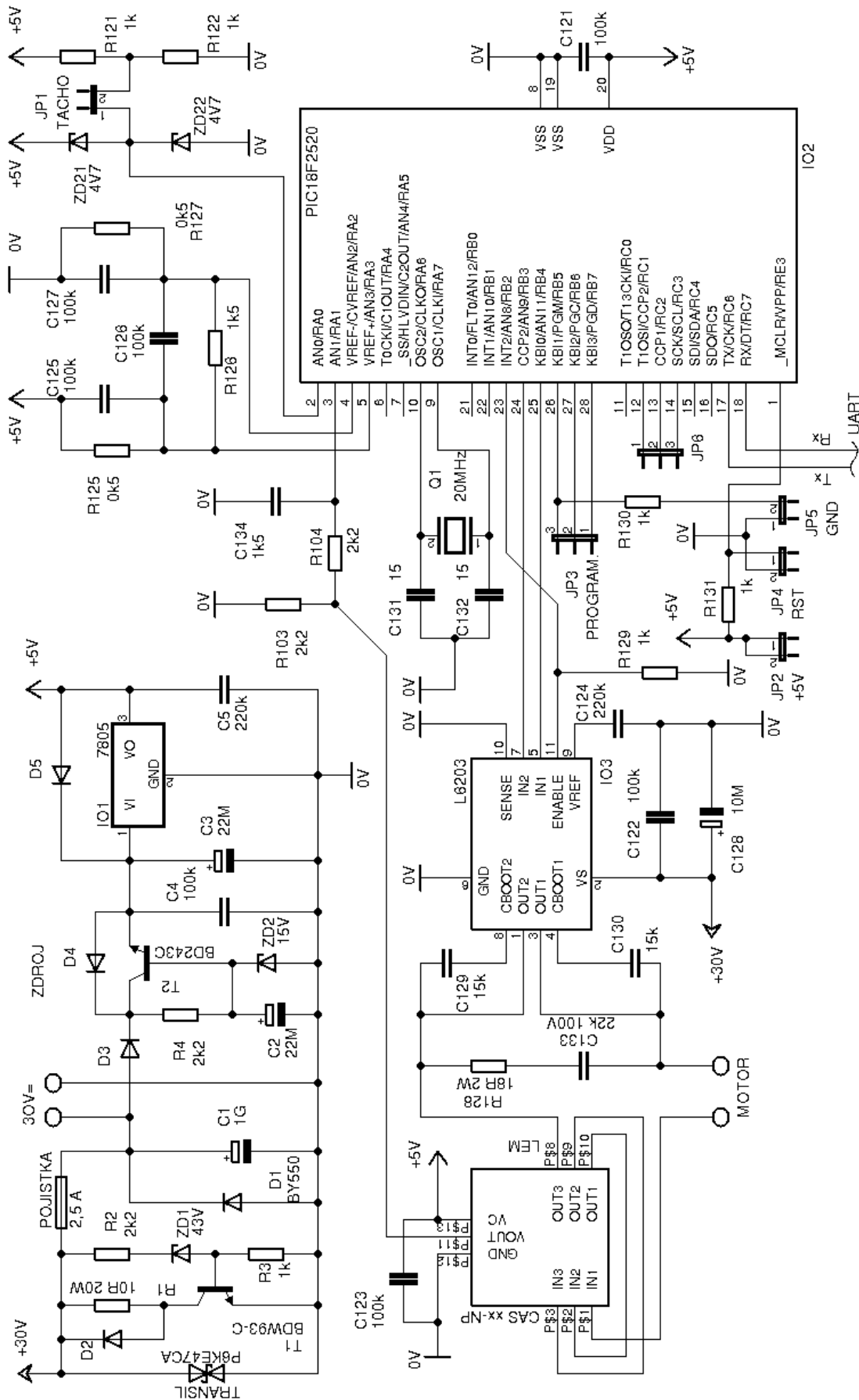


obr.6.1. Blokové schéma

K získání napětí 5 V pro řídicí elektroniku jsou použity dva kaskádně zapojené lineární stabilizátory napětí s tranzistorem a integrovaným obvodem 7805. Pro obousměrnou komunikaci mikrokontroléru s počítačem je možné využít opticky oddělenou sériovou linku UART s převodníkem na rozhraní USB. Způsobem komunikace s počítačem se podrobně zabývá zdroj [3], odkud je převzato u její elektrické zapojení.

### 6.2 Elektrické schéma

Úplné schéma je na obr. 6.2 (měnič) a 6.3 (komunikace přes USB). Diody jsou typu 1N4007 SMD, pokud není uvedeno jinak.



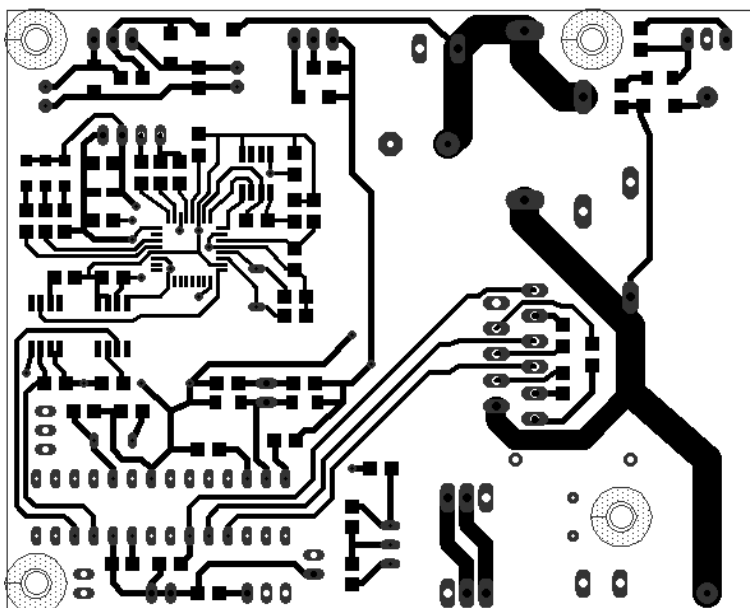
obr.6.2. Úplné schéma (1/2) - měnič



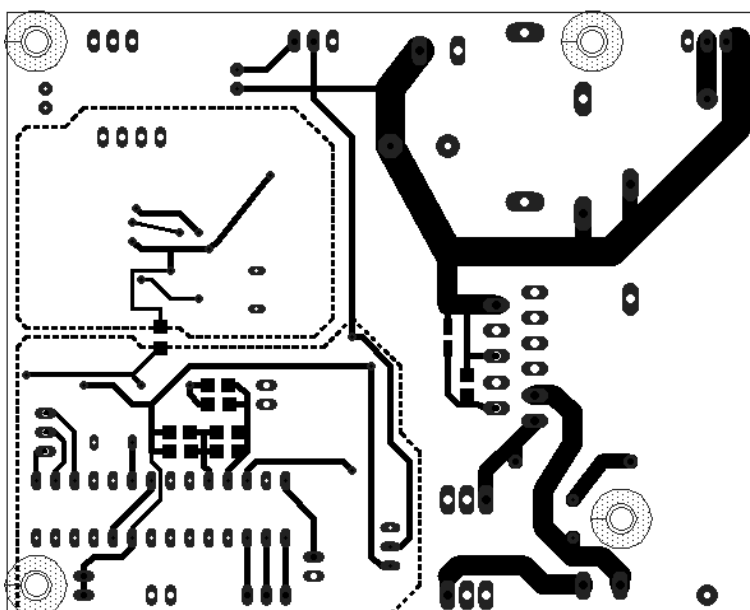


### 6.3 Plošný spoj

Motiv desky plošných spojů (DPS) vytvořený v programu Eagle je na obr. 6.3 a 6.4. Všechny obrázky desky jsou z pohledu zepředu.



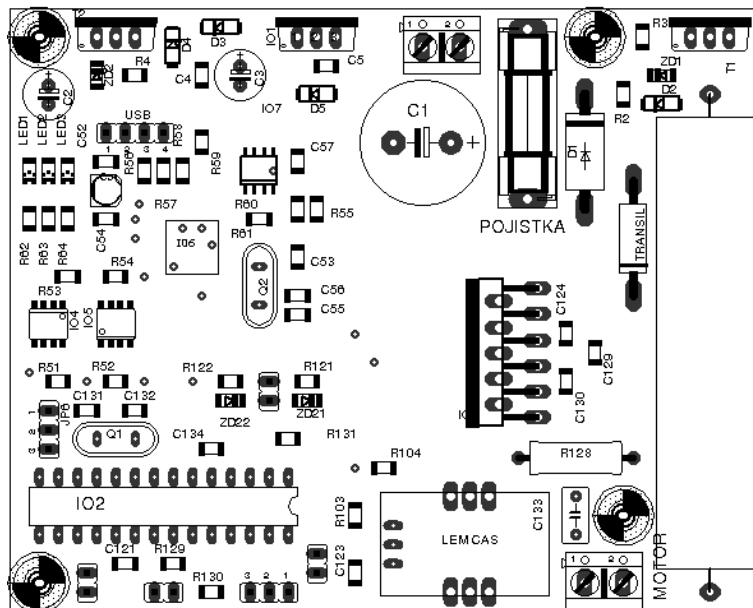
obr.6.4. Motiv DPS 1:1 - přední strana



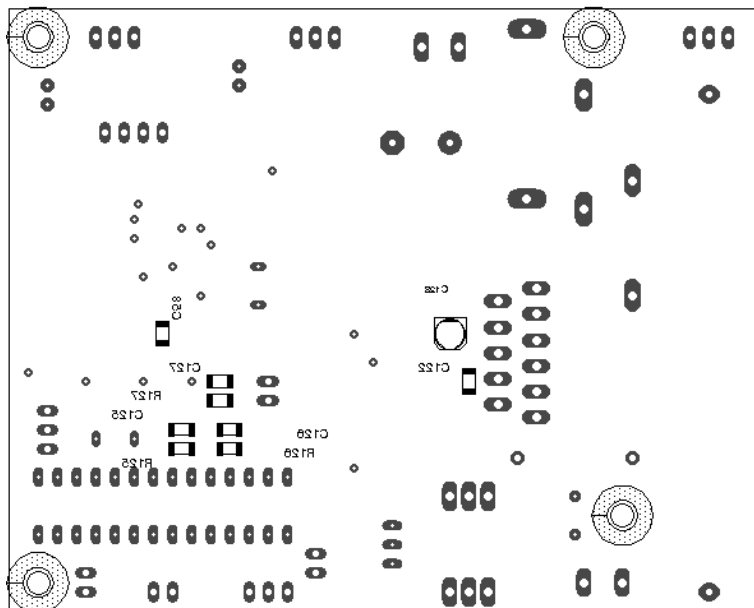
obr.6.5. Motiv DPS 1:1 - zadní strana

## 6.4 Osazovací plán

Osazovací plán k DPS viz obr. 6.6 a 6.7.



obr.6.6. Osazovací plán DPS - přední strana



obr.6.7. Osazovací plán DPS - zadní strana

## 6.5 Seznam součástek

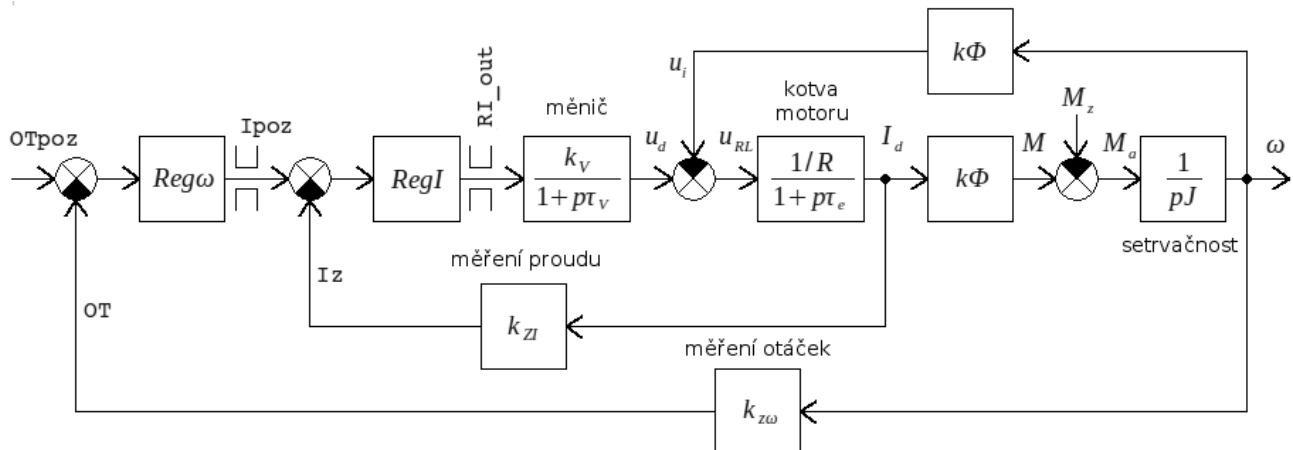
tab.6.1. Seznam součástek

Integrované obvody		
počet	typ	pouzdro
1	7805	TO-220
1	PIC18F2520	DIP28
1	L6203	Multiwatt11
1	LEM CAS 15-NP	
2	HCPL0600	SOIC08
1	FT232BL	LQFP32
1	93-C46	SO-08
Tranzistory		
počet	typ	pouzdro
1	BDW93C	TO-220
1	BD243C (BDW93C)	TO-220
Diody, Zenerovy diody, transily		
počet	typ	pouzdro
4	1N4007 (SMD)	DO-214AC
1	BY550	
1	Zener. 43V	SOD80C
1	Zener. 15V	SOD80C
1	transil P6KE47CA	DO-15
2	Zener. 4V7	SOD80C
Ostatní		
počet	typ	pouzdro
1	krystal 6MHz	HC49U
1	krystal 20MHz	HC49U
3	LED SMD	1206
1	kolíková lišta	
1	patice DIP28	
2	svorkovnice	PTR AK500/2

Odpory		
počet	hodnota	pouzdro
2	27R	1206
1	100R	1206
1	470	1206
17	1k	1206
2	1k5	1206
4	2k2	1206
2	10k	1206
1	10R 20W	drátový
1	18R 2W	
Kondenzátory		
počet	hodnota	pouzdro
2	15p	1206
2	27p	1206
1	1n	1206
1	1n5	1206
2	15n 50V	1206
1	33n	1206
10	100n	1206
2	220n	1206
1	22n 100V fóliový	rozteč 5 mm
1	1 $\mu$ , 35V, tantal, SMD	B
1	10 $\mu$ 63V 105°C, elyt.	rozteč 2,5 mm
2	22 $\mu$ 63V 105°C, elyt.	rozteč 2,5 mm
Mechanické konstrukční prvky		
počet	typ	
3	chladič pro TO-220	
1	chladič pro Multiwatt11	
4	šroub M3x6	
4	distanční sloupek M3x20	

## 7. Regulace stejnosměrného motoru

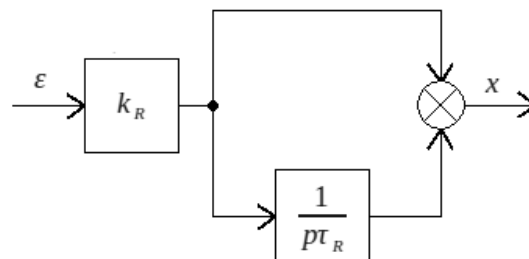
### 7.1 Schéma regulace



obr.7.1. Blokové schéma regulace otáček

Blokové schéma regulace otáček je na obr. 7.1. Jelikož jsou oba regulátory realizovány číslicově v mikrokontroléru, veličiny  $OT_{poz}$ ,  $I_{poz}$ ,  $RI\_out$ ,  $I_z$  a  $OT$  jsou bezrozměrná čísla odpovídající názvům jednotlivých proměnných v programu. Bloky „měření proudu“ a „měření otáček“ v sobě zahrnují jak příslušné čidlo (převodník proudu nebo otáček na napětí), tak i A/D převod realizovaný v mikrokontroléru.

### 7.2 Řídicí algoritmus



obr.7.2. Analogový PI-regulátor

Činnost PI-regulátoru lze popsat blokovým schématem na obr. 7.2. Závislost výstupu  $x$  na regulační odchylce  $\varepsilon$  se dá vyjádřit buď přenosem v Laplaceově transformaci:

$$F(p) = k_R \cdot \frac{1 + p\tau_R}{p\tau_R} = k_R \cdot \left(1 + \frac{1}{p\tau_R}\right) \quad (7.1)$$

nebo pro časové průběhy pomocí rovnice:

$$x(t) = k_R \cdot \varepsilon(t) + \frac{k_R}{\tau_R} \cdot \int_0^t \varepsilon(t) dt \quad (7.2)$$

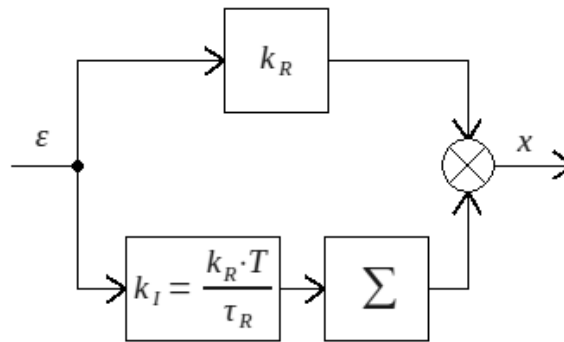
kde  $\varepsilon$  je regulační odchylka,  $k_R$  a  $\tau_R$  jsou konstanty regulátoru a  $x$  je výstup.

Číslicovou realizací PI-regulátoru vznikne tzv. PS-regulátor. Ten se od PI-regulátoru liší mimo jiné tím, že nepracuje spojitě v čase, ale výpočet výstupní hodnoty  $x$  probíhá v určitých

diskrétních časových okamžicích s periodou  $T (=T_{PWM})$ . Čas je udáván v celistvých násobcích této periody ( $nT$ ) a integrál je rovněž počítán numericky a přechází na sumu:

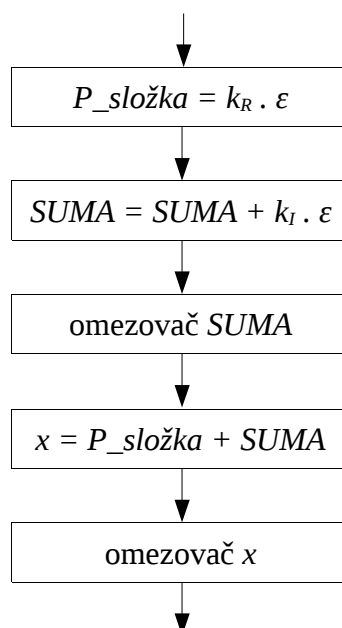
$$x(nT) = k_R \cdot \varepsilon(nT) + \frac{k_R}{\tau_R} \cdot \sum_{i=0}^{n-1} T \cdot \varepsilon(iT) \quad (7.3)$$

Pro co možná nejmenší výpočtovou náročnost (program má běžet na 8-bitovém procesoru, který nemá instrukce pro výpočty s desetinnými čísly či dokonce s plovoucí čárkou, a i jednoduché operace s 16-bitovými celými čísly mohou znamenat desítky instrukcí) bylo použito blokové schéma PS-regulátoru podle obr. 7.3.



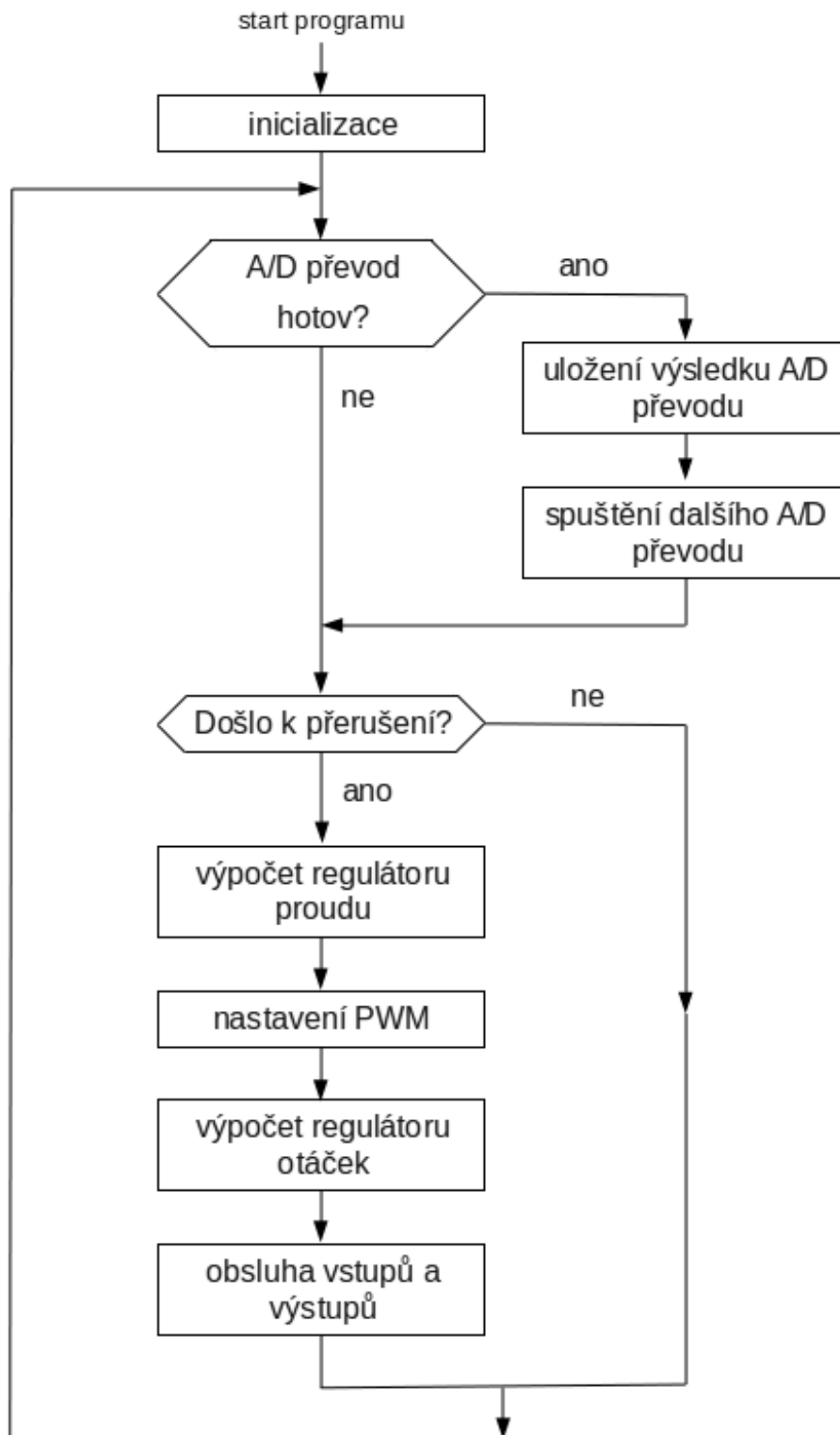
obr.7.3. Blokové schéma použitého PS-regulátoru

Výraz  $\frac{k_R \cdot T}{\tau_R}$  byl shrnut do jedinné konstanty  $k_I$ , která určuje rychlost integrace (tato konstanta vychází většinou menší než 1, takže v programu ve skutečnosti probíhá dělení její převrácenou hodnotou). Nevýhoda tohoto řešení je zřejmá – při každé změně nastavení regulátoru (konstant  $k_R$  a  $\tau_R$ ) je třeba vypočítat novou hodnotu konstanty  $k_I$ . Blok  $\Sigma$  zajišťuje samotnou numerickou integraci, tj. při každém proběhnutí výpočtu seče vstupní hodnotu se svojí výstupní hodnotou z minulého výpočtu. Každý výpočet PS-regulátoru probíhá podle vývojového diagramu na obr. 7.4. Stav integrátoru  $SUMA$  a výstup regulátoru  $x$  jsou po svém vypočítání vždy omezeny, aby nedocházelo přehlcení integrátoru nebo vstupů dalších bloků.



obr.7.4. Vývojový diagram PS-regulátoru

Popsaný PS-regulátor je použit ve stejné podobě pro regulaci proudu i otáček. Celý řídicí algoritmus je popsán blokovým schématem na obr. 7.5. Program se hned po inicializaci dostane do nekonečné smyčky. S periodou  $T=T_{PWM}$  dochází k přerušení (přetečení čítače TIMER0), které následně vyvolá proběhnutí výpočtu obou regulátorů, přičemž přednost má regulátor proudu, který musí reagovat s co nejmenším dopravním zpožděním. A/D převod probíhá neustále, pokud program vyhodnotí jeho dokončení, uloží změřené hodnoty a spustí další A/D převod, přičemž se střídavě převádí proud a otáčky.



obr.7.5. Vývojový diagram řízení motoru

### 7.3 Proudová smyčka

Pro výpočet regulační smyčky proudu je potřeba znát především elektrické parametry motoru, přenos měniče a čidla proudu. Do přenosu měniče je zahrnuto nejen jeho statické „zesílení“  $k_V$ , ale i dopravní zpoždění dané jeho časově nespojitým řízením.

Odpor vinutí kotvy byl změřen Ohmovou metodou, indukčnost kotvy pak pomocí osciloskopu na motoru připojeném k měniči. Naměřené hodnoty jsou:

$$R_A = 1,5 \Omega$$

$$L_A = 1,9 \text{ mH}$$

Z toho lze získat časovou konstantu kotvy  $\tau_e$  podle vztahu:

$$\tau_e = \frac{L_A}{R_A} = \frac{1,9 \cdot 10^{-3}}{1,5} = 1,26 \cdot 10^{-3} \text{ s} \quad (7.4)$$

Dopravní zpoždění vlivem A/D převodu proudu, výpočtu regulátoru proudu a zpoždění pulsně šířkové modulace se pohybuje v rozmezí 1 až 2  $T_{PWM}$  (podle nastavené střídy PWM), průměrné dopravní zpoždění je tedy 1,5  $T_{PWM}$ . Dopravní zpoždění lze tedy aproximovat aperiodickým blokem s časovou konstantou:

$$\tau_V = 1,5 T_{PWM} = 1,5 \cdot 2,048 \cdot 10^{-4} = 3,072 \cdot 10^{-4} \text{ s} \quad (7.5)$$

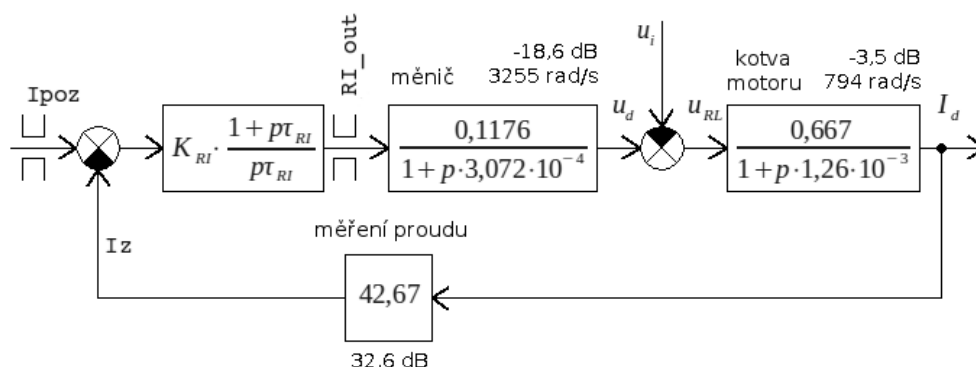
Konstanta měniče  $k_V$  při maximálním výstupním napětí měniče 30 V a 256 úrovněmi střídy PWM pro každou polaritu vychází jako:

$$k_V = \frac{30}{255} = 0,1176 \quad (7.6)$$

Čidlo proudu má převod 0,625 V / 5 A. A/D převodník má celkem 1024 úrovně a rozdíl jeho referenčních napětí je 3 V, takže jeho převod je 1024 / 3V. Celkový převod bloku „měření proudu“ tedy je:

$$k_{ZI} = \frac{0,625}{5} \cdot \frac{1024}{3} = 42,67 \quad (7.7)$$

Na obr. 7.6 je blokové schéma proudové smyčky s přenosy jednotlivých bloků. Kromě toho je u bloků uvedena jejich mezní frekvence a přenos při  $\omega = 0$  v dB.

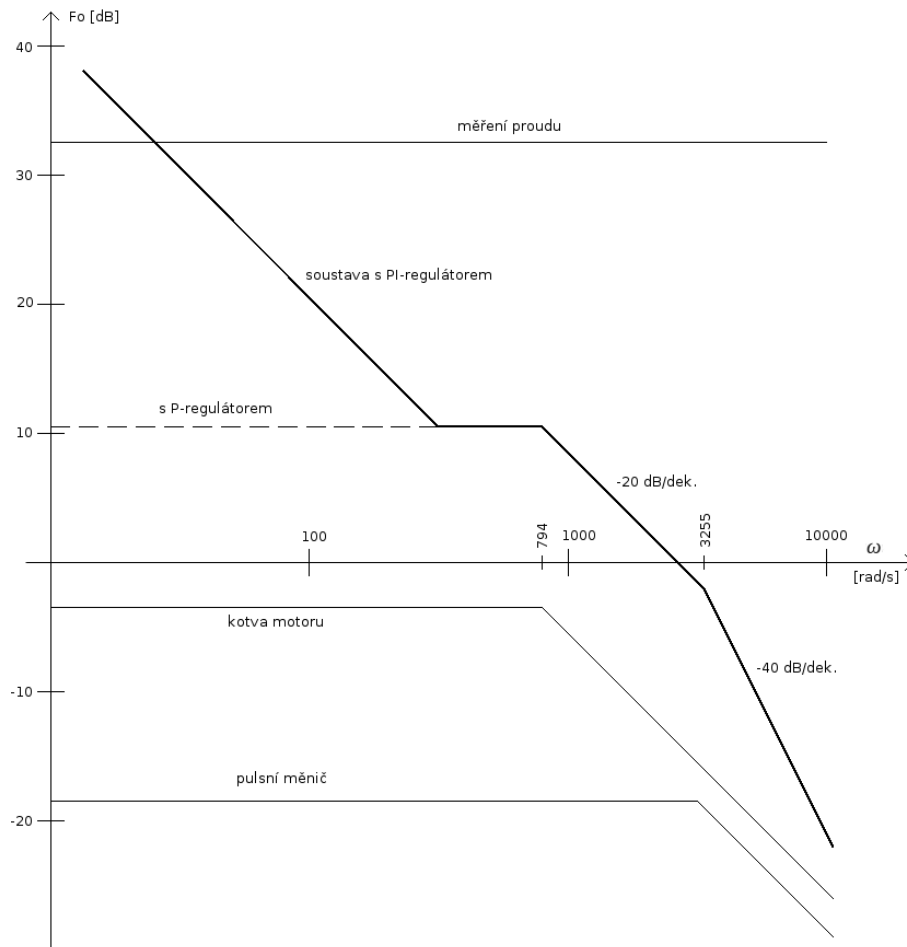


obr.7.6. Proudová regulační smyčka

Příklad amplitudové frekvenční charakteristiky je na obr. 7.7. Graf udává zesílení otevřené proudové smyčky při nastavení regulátoru  $k_R = 1$  a  $k_I = 1/16$  ( $\tau_R = 3,16 \cdot 10^{-3}$  s). Fázový posuv otevřené smyčky při  $\omega = \omega_f$  vychází:

$$\varphi(\omega_f) = -\arctan\left(\frac{3100}{3255}\right) - \arctan\left(\frac{3100}{3255}\right) + \arctan\left(\frac{3100}{316}\right) - 90^\circ = -125^\circ$$

Bezpečnost ve fázi tak je pouze 55°, takže skutečné zesílení je třeba nastavit přibližně o 5 dB nižší.



obr.7.7. Příklad frekvenční amplitudové charakteristiky otevřené proudové smyčky

Podobným způsobem jako u proudového regulátoru, lze spočítat i nastavení regulátoru otáček. Jelikož ale neznáme moment setrvačnosti motoru, bylo nastavení tohoto regulátoru provedeno experimentálně.



## 8. Programové řešení

Řídící program byl napsán v jazyce C ve vývojovém MPLAB® od společnosti Microchip za použití překladače C18 Compiler od stejné firmy. Zde je uveden příklad řídicího programu po jednotlivých blocích. Požadované otáčky jsou v tomto případě dány rampou. Program lze tedy doplnit například a nastavování otáček 2 tlačítky nebo po sériové lince UART.

Direktivy překladače, obsluha přerušení:

```
#include <p18f2520.h>
#include <delays.h>
#pragma config CCP2MX = PORTBE           // připojí výstup CCP2 na bránu B (RB3)
#pragma config OSC = HS                   // high-speed oscilátor s krystalem
#pragma config PBDEN = OFF                // port B jako digitální I/O

static unsigned char Cas = 0, tik=0;

#pragma code high_vector=0x08

void interrupt_at_high_vector (void)
{
    if (INTCON & 0x04)                    // preruseni od timeru 0
    {
        Cas++;                            // je třeba provést výpočet regulátorů
        INTCON &= ~0x04;                 // vynulovani priznaku preruseni
    }
    INTCON |= 0x80;                       // opetovne povoleni vseh preruseni
}
```

Deklarace proměnných a inicializace:

```
void main (void)
{
    // měření
    static unsigned char Admux=1;          // 1-proud, 0-otáčky
    static unsigned char ADL=0, ADH=0;
    static unsigned int Iad=0, OTad=0;
    static signed int Iz = 0, OT=0;
    // Regulátor otáček
    static signed int ROT_E = 0, ROT_Pk = 4, ROT_I = 0, ROT_P = 0, ROT_out = 0, OTpoz = 50;
    static signed char ROT_integL = 0;
    static signed int OToffset = 0;
    // Regulátor proudu
    static signed int RI_E = 0, RI_Pk = 2, RI_I = 0, RI_P = 0, RI_out = 0, Ipoz = 10;
    static signed char RI_integL = 0;
    static signed int Ioffset = -8;        // chyba nuly při měření proudu
    // pomocné
    static signed int x=0, i=0;
    static unsigned char poll1=0, poll2=0, T=0, smer=1;

    // inicialice portů, ochrana před spuštěním měniče
    TRISB = 0xFF;                          // Nastaveni portu B jako vstup (vysoka impedance)
    TRISA = 0xFF;                          // Nastaveni portu A jako vstup (vysoka impedance pro A/D vstupy)
    TRISC = 0xFF;                          // Nastaveni portu C jako vstup (pro jistotu)
    PORTB = 0x00;                          // Vynulovani portu B
    Delay1KTCYx(250);                       // ochranna prodleva 50 ms (20MHz)
}
```

## Konfigurace modulů a periférií mikrokontroléru:

```

// nastaveni TIMER0
T0CON = 0xC1;      // timer0, preddelicka deli 4 (20 MHz)
INTCON = 0xA0;    // nastaveni preruseni

// nastaveni PWM
T2CON = 0x05;     // postscale 1:1, TIMER2 zapnuty, preddelicka deli 4
PR2 = 0xFF;      // nastaveni periody PWM

CCP2CON = 0x0C;   // vybere PWM modul a nastavi nejnizsi 2 bity PWM stridy na 00
                // (PWM je 10-bitove)
CCPR2L = 0x00;   // nastavi stridu PWM na 0
TRISB = 0xE0;    // Nastaveni RB0-RB4 jso vystup

// nastaveni A/D prevodniku
ADCON1 = 0x3B;   // reference Vref+, Vref-
ADCON0 = 0x04;   // kanal A1
ADCON2 = 0x95;   // Tad = 16Tosc , Tacq = 4Tad, zarovnani doprava (20 MHz)
ADCON0 = 0x05;   // kanal AN1, zapni

// ostatni nastaveni, signalizace
TRISC=0x00;     // nastaveni portu C jako vystupu (signalizacni LEDky)
PORTC = 0x00;   // vynulovani portu C

PORTB = 0x04;   // nastaveni měniče ENABLE = 1, POLARITA = 0
ADCON0 = 0x07;   // spust A/D převodník

```

## Obsluha A/D převodníku a ukládání naměřených hodnot:

```

while(1)
{
    if (!(ADCON0 & 0x02))          // DOKONCEN A/D PREVOD
    {
        PIR1 = (PIR1 & (~0x40));  // vynuluje ADIF
        ADH = ADRESH;
        ADL = ADRESL;

        // multiplex proud/otacky
        if (Admux)                // mereni proudu
        {
            ADCON1 = 0x0B;        // v pristim cyklu
            ADCON0 = 0x03;        // se meri otacky

            Iad = ((unsigned int) (ADH)) << 8 ;
            Iad += (unsigned int) ADL;
            Admux = 0;
        }
        else                      // mereni otacek
        {
            ADCON1 = 0x3B;        // v pristim cyklu
            ADCON0 = 0x07;        // se meri proud

            OTad = ((unsigned int) (ADH)) << 8 ;
            OTad += (unsigned int) ADL;
            Admux = 1;
        }
    }
}

```

## Regulátor proudu:

```
// VYPOCET REGULATORU (pretečení TMR0)
if (Cas)
{
    Cas--;

    // REGULATOR PROUDU
    Iz = 512 - (signed int) Iad;           // "oznamenkovani" proudu
    Iz += Ioffset;
    RI_E = Ipoz - Iz;                     // regulační odchylka

    // RegI - P složka
    RI_P = RI_Pk * RI_E;                  // proporcionální složka

    // RegI - integrace
    RI_integL += RI_E;                    // integrace
    if (RI_integL > 3)                     // snížení rychlosti integrace
    {
        RI_I+=1;
        RI_integL = 0;
    }
    if (RI_integL < -3)
    {
        RI_I-=1;
        RI_integL = 0;
    }

    if (RI_I > 255) RI_I = 255;            // omezovač integrátoru
    if (RI_I < -255) RI_I = -255;

    // RegI - vystup
    RI_out = RI_P + RI_I;                  // součet proporcionální a integrační složky

    if (RI_out > 255) RI_out = 255;        // omezovač vystupu regulátoru
    if (RI_out < 0) RI_out = 0;

    // RIZENI MENICE
    if (RI_out==0)
        CCPR2L=0;

    if (RI_out > 0)
    {
        PORTB = 0x04; // polarita=0, enable=1
        CCPR2L = (unsigned char) RI_out;
    }
    if (RI_out < 0)
    {
        PORTB = 0x14; // polarita=1, enable=1
        CCPR2L = (unsigned char) (255 + RI_out);
    }
}
```

## Regulátor otáček:

```

// REGULATOR OTACEK
OT = -512 + (signed int) OTad;      // "oznamenkovani" otacek
OT += OToffset;
ROT_E = OTpoz - OT;                // regulační odchylka

// RegI - P složka
ROT_P = ROT_Pk * ROT_E;            // proporcionální složka

// RegOT - integrace
ROT_integL += ROT_E;               // integrace
if (ROT_integL > 63)                // snížení rychlosti integrace
{
    ROT_I+=1;
    ROT_integL = 0;
}
if (ROT_integL < -63)
{
    ROT_I-=1;
    ROT_integL = 0;
}
if (ROT_I > 80) ROT_I = 80;         // omezovač integrátoru
if (ROT_I < -80) ROT_I = -80;

// RegOT - vystup
ROT_out = ROT_P + ROT_I;           // součet proporcionální a integrační složky

if (ROT_out > 80) ROT_out = 80;    // omezení požadovaného proudu
if (ROT_out < -80) ROT_out = -80;

Ipoz = ROT_out;                    // nastavení požadovaného proudu

```

## Nastavení požadovaných otáček (rozběh a zastavení po rampě) a konec smyčky:

```

T++;
if (T>250)
{
    T=0;
    if(smer)
        OTpoz++;
    else
    {
        OTpoz--;
        OTpoz--;
        OTpoz--;
    }
    if (OTpoz>300)
        smer=0;
    if (OTpoz<2)
        smer=1;
}
// vynulovani WATCHDOGU
_asm
CLRWDI
_endasm
}
}

```

## 9. Závěr

Výsledkem této práce je funkční zařízení, které je schopné při napájení z laboratorního zdroje řídit otáčky malého stejnosměrného motoru s permanentními magnety. Zařízení bylo testováno na stroji o výkonu řádově stovek Wattů. Během této práce jsem si v praxi ověřil teoretické znalosti z několika různých oblastí, od elektrických strojů přes elektroniku až po programování. Zatímco elektrické zapojení se osvědčilo, u řídicího programu se počítá se dalším vývojem, zejména co se týče komunikace zařízení s obsluhou nebo s počítačem a tím i možného nastavování parametrů regulátoru beze změny programu.

## Seznam použité literatury

### Knihy a skripta

- [1] Pittermann, M.: Elektrické pohony, základy. Skripta ZČU, Plzeň 2008
- [2] Vondrášek, F.: Výkonová elektronika. Svazek 3. Skripta ZČU, Plzeň 2003
- [3] Dolejší, M: Sériová komunikace mikrokontroléru PIC a PC. Bakalářská práce. Západočeská univerzita v Plzni, Fakulta elektrotechnická. Plzeň 2010

### Elektronické a webové zdroje

- [W1] katalogový list obvodu L6203 fy. SGS-Thomson Microelectronics. Červenec 1997
- [W2] katalogový list mikrokontroléru PIC18F2520 od fy. Microchip. 2007
- [W3] katalogový list obvodu CAS 15-NP od fy. LEM. [www.lem.com](http://www.lem.com), 8.březen 2012