

Univerzita Pardubice
Dopravní fakulta Jana Pernera

Řídicí jednotka pro čtyřkvadrantový měnič pro DC
motory s permanentními magnety

Bc. Michal Foltýn

Diplomová práce

2008

Univerzita Pardubice
Dopravní fakulta Jana Pernera
Katedra elektrotechniky, elektroniky a zabezpečovací techniky v dopravě
Akademický rok: 2007/2008

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Michal FOLTÝN**
Studijní program: **N3708 Dopravní inženýrství a spoje**
Studijní obor: **Dopravní infrastruktura-Elektrotechnická zařízení v dopravě**
Název tématu: **Řídicí jednotka pro čtyřkvadrantový měnič pro DC motory s permanentními magnety**

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

1. Návrh koncepce řídicí jednotky a volba konkrétního typu mikrokontroléru řady AVR
2. Návrh obvodového řešení řídicí jednotky pro měnič s mikrokontrolérem řady AVR
3. Konstrukce, oživení a odzkoušení řídicí jednotky.
4. Sestavení řídicího SW pro mikrokontrolér v řídicí jednotce umožňující řízení pohonu s DC motorem ve všech čtyřech kvadrantech

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná**

Seznam odborné literatury:

- [1] JAVŮREK, Jiří. Regulace moderních elektrických pohonů. Praha : Grada Publishing a.s., 2003. 262 s. ISBN 80-247-0507-9.
- [2] Tovární dokumentace k použitým součástkám

Vedoucí diplomové práce:

Ing. Jiří Šlmánek

Katedra elektrotechniky, elektroniky a zabezpečovací techniky v dopravě

Datum zadání diplomové práce:

11. prosince 2007

Termín odevzdání diplomové práce:

4. června 2008

prof. Ing. Bohumil Culek, CSc.

děkan

L.S.

prof. Ing. Vladimír Schejba, CSc.

vedoucí katedry

V Pardubicích dne 25. února 2008

PODĚKOVÁNÍ

Na tomto místě bych chtěl bych poděkovat Ing. Jiřímu Šimánkovi, bez jehož pomoci, cenných rad a připomínek by tato práce nevznikla. Další díky patří Ing. Zdeňku Maškovi za jeho pomoc při tvorbě softwaru. Vděk patří také všem, kteří mě dokázali přivést na jiné myšlenky a nabrat tak novou sílu pro další tvorbu . . .

SOUHRN

Diplomová práce se zabývá kompletní realizací řídicí jednotky s jednočipovým mikrokontrolérem pro ovládání čtyřkvadrantového měniče pro DC motory s permanentními magnety. Zaměřuje se zejména na softwarovou realizaci regulačního obvodu. Na konci práce je popsáno testování zkonstruované jednotky

KLÍČOVÁ SLOVA

čtyřkvadrantové měniče; mikrokontroléry; PS regulátory; PWM modulátory; regulační obvody

TITLE

Control unit for four-quadrant converter for DC motors with permanent magnets

ABSTRACT

The Thesis deals with complete realization of control unit with one-chip microcontroller for driving of four-quadrant converter for DC motors with permanent magnets. It focuses especially on software realization of control circuit. The testing of construct unit is described in the end of the thesis.

KEYWORDS

four-quadrant converters; microcontrollers; PS controllers; PWM modulators; control circuits

OBSAH

ÚVOD.....	8
1 KONCEPCE ŘÍDÍCÍ JEDNOTKY	10
1.1 Čtyřkvadrantový pulzní měnič	10
1.2 Sériová komunikace	13
1.3 IRC čidlo	14
1.4 Měření proudu a napětí.....	15
2 VÝBĚR SOUČÁSTEK A OBVODOVÝ NÁVRH.....	17
2.1 Blokové schéma.....	17
2.2 Výběr součástek.....	18
2.2.1 Mikrokontrolér	18
2.2.2 Displej.....	19
2.2.3 Ostatní obvody.....	20
2.3 Obvodový návrh	21
2.3.1 Napájení.....	21
2.3.2 Zpracování proudu a napětí	23
2.3.3 Zpracování signálu z IRC čidla	24
2.3.4 LCD panel a ovládání	25
2.3.5 Obvody mikrokontroléru a další obvody.....	25
2.4 Provedení řídicí jednotky	27
3 NASTAVENÍ REGISTRŮ MIKROKONTROLÉRU	28
3.1 Časovače/čítače	28
3.1.1 Timer/Counter0	28
3.1.2 Timer/Counter1	30
3.1.3 Timer/Counter3	31
3.2 Sériový port	32
3.3 Další periférie	34
4 OBSLUHA PERIFERNÍCH ZAŘÍZENÍ.....	37
4.1 Zobrazovací jednotka	37
4.2 Komunikační protokol.....	39
4.3 Vstupní obvody	41

5	REGULAČNÍ STRUKTURA	42
5.1	Regulační obvod	42
5.2	PS Regulátor	44
5.3	PWM modulátor	46
6	VÝSLEDNÝ ŘÍDÍCÍ SOFTWARE	48
6.1	Hlavní smyčka	48
6.2	Vedlejší smyčky	49
6.3	Činnost programu v přerušení	50
7	TESTOVÁNÍ ŘÍDÍCÍ JEDNOTKY	52
7.1	Testovací pracoviště	52
7.2	Regulace proudu	54
7.3	Regulace otáček	55
7.4	Regulace napětí	56
	ZÁVĚR	57
	POUŽITÁ LITERATURA	58
	SEZNAM TABULEK	59
	SEZNAM OBRÁZKŮ	60
	SEZNAM ZKRATEK	61
	SEZNAM PŘÍLOH	62

ÚVOD

Přestože v dnešní době asynchronní motory vytlačují stejnosměrné motory, nacházejí se stále odvětví, v nichž má stejnosměrný motor uplatnění. Stejnosměrné (neboli DC) motory s permanentními magnety jsou v současné době používány tam, kde není požadován příliš velký výkon. Jedná se tak například o pohony napájené z baterií nebo o různé modelářské pohony. Jejich výhodou je, že mají vlastní buzení a jejich konstrukce je tak jednodušší.

Regulovat DC motory s permanentními magnety lze různými způsoby, od nejjednodušších (např. změna napětí) až po složitější, které generují vhodně upravený signál a jsou vybaveny zpětnou vazbou. O volbě regulátoru a tedy i řízení motoru rozhoduje to, k jakému účelu bude motor využit.

Cílem diplomové práce je navrhnout, zkonstruovat, oživit a odzkoušet řídicí jednotku s jednočipovým mikrokontrolérem pro řízení stejnosměrného motoru s permanentními magnety. Koncepce řídicí jednotky bude vhodnými obvody a algoritmy zpracovávat vstupní i výstupní data tak, aby docházelo k co neefektivnějšímu využívání dostupného výpočetního výkonu mikrokontroléru. Výsledkem by měla být jednotka umožňující univerzální využití.

Obsluha bude schopna ovládat měnič pomocí řídicí jednotky dvěma způsoby. Řízení bude možné přímo na řídicí jednotce nebo prostřednictvím počítače. Počítač bude připojen k řídicí jednotce přes sériovou sběrnici. Komunikace bude zajišťovat parametrizování jednotky, načítání dat v reálném čase a samozřejmě i zadávání žádané hodnoty.

Řídicí jednotka bude zajišťovat řízení stejnosměrného motoru v otáčkové nebo proudové smyčce. Při řízení v proudové smyčce lze říci, že se jedná též o momentovou smyčku, protože moment lineárně stoupá nebo klesá s proudem. Potřebné data k řízení budou získávána z inkrementálního čidla (otáčky), z čidel pro měření proudu, případně i napětí. Řídicí jednotka by měla být schopna zvládnout provoz stejnosměrného motoru ve dvou/čtyřech kvadrantech, tj. motorický a generátorický režim v obou směrech. Jednotka bude také vybavena displejem pro zobrazování požadované a skutečné hodnoty regulované veličiny.

Podle zjištěných vstupních a požadovaných dat lze použitím vhodného regulátoru nastavovat výstupní PWM (Pulse Width Modulation) signál, který je generován přímo v mikrokontroléru. Pro řízení stejnosměrného motoru s permanentními magnety je vhodným typem regulátor PS (Proporcionálně Sumační) – diskretní obdoba analogového PI (Proporcionálně Integračního) regulátoru.

V následujících kapitolách bude popsáno konstrukční řešení a programové vybavení. Kapitola 1 je věnována popisu zařízení, se kterými bude řídicí jednotka spolupracovat. Další kapitola se zabývá výběrem vhodných součástek a samotným obvodovým návrhem a konečnou realizací. Kapitola 3 popisuje registry periférií mikrokontroléru, jež byly pro řídicí jednotku uplatněny. Další 3 kapitoly se zabývají nejprve obsluhou displeje, načítáním vstupních dat a komunikací, poté regulačními strukturami a konečným programem. Poslední kapitola popisuje testování řídicí jednotky.

1 KONCEPCE ŘÍDÍCÍ JEDNOTKY

Základem pro návrh řídicí jednotky je to, s jakými dalšími zařízeními bude jednotka spolupracovat. Je vhodné seznámit se s jejich principem a z toho vyplývajícími nároky na řídicí jednotku. Jedná se zejména o čtyřkvadrantový pulzní měnič (již zkonstruován včetně budících obvodů – není součástí práce), způsob komunikace s počítačem po sériové lince, snímač otáček (IRC čidlo) a snímače pro měření proudu a napětí.

Dle seznámení s těmito zařízeními je pak možné určit, jaká data a v jakém tvaru je bude jednotka přijímat nebo odesílat, zda-li bude potřeba dalších speciálních zařízení nebo bude postačovat běžné vybavení mikrokontroléru.

1.1 Čtyřkvadrantový pulzní měnič

K řízení střední hodnoty stejnosměrného napětí jsou určeny pulzní měniče. Na vstupu pulzního měniče je konstantní stejnosměrné napětí a na jeho výstupu je napětí s říditelnou střední hodnotou. Pulzní měniče se používají tam, kde je k dispozici stejnosměrná napájecí síť a současně je třeba řídit výstupní napětí, např. pro napájení rotorového obvodu (kotvy) stejnosměrného stroje, nebo zajištění předstabilizace napětí stejnosměrného meziobvodu. Převážně se tak jedná o trakční pohony nebo pohony napájené z akumulátorů. V pohonech menších výkonů jsou pulzní měniče také často používány pro svoji jednoduchost.

Střední hodnota napětí je dána výrazem:

$$U_{dAV} = \frac{1}{T} \int_{t=0}^{t_z} U_1 \cdot dt = U_1 \cdot \frac{t_z}{T} = U_1 \cdot z, \text{ kde} \quad (1)$$

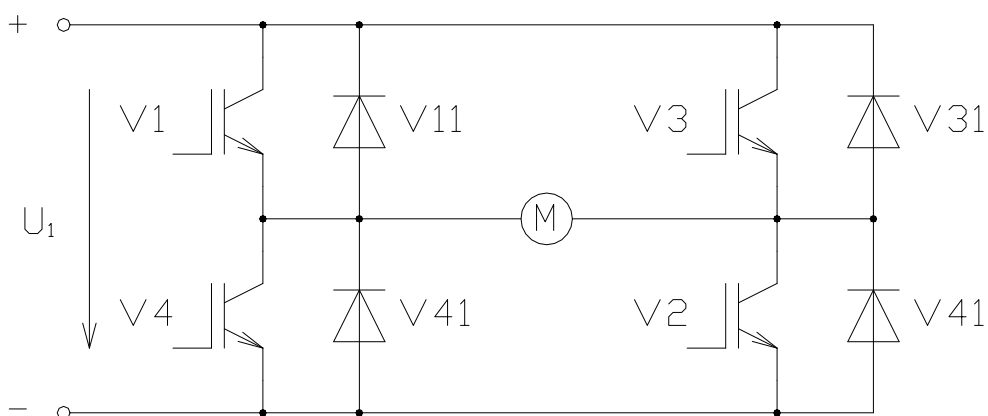
t_z – doba zapnutí pulzního měniče

T – perioda

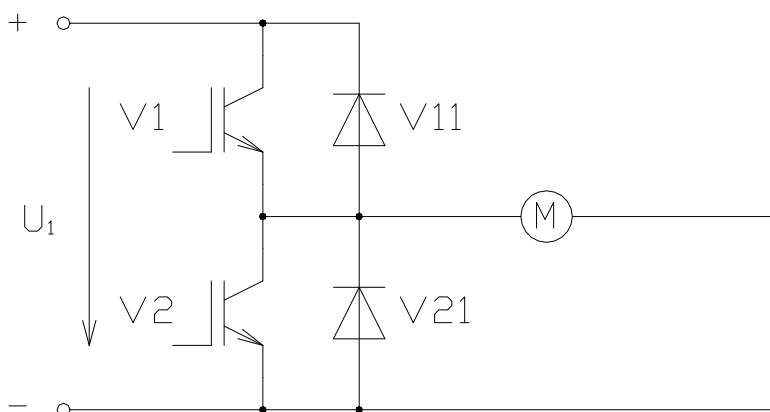
$z = \frac{t_z}{T}$ – zatěžovatel.

Zatěžovatel se může pohybovat od 0 do 1, výstupní napětí se pak mění od 0 do U_1 . Střední hodnota výstupního napětí pulzního měniče je tedy přímo úměrná velikosti zatěžovatele. Spínací kmitočet (převrácená hodnota periody T) se obvykle pohybuje od stovek Hz do jednotek kHz. Velikost závisí zejména na druhu použité výkonové spínací (polovodičové) součástky.

Aby bylo možné provozovat pulzní měnič ve všech čtyřech kvadrantech, tj. s oběma polaritami napětí a proudu, používá se čtyřkvadrantový pulzní měnič (H-můstek), kde se přechod mezi jednotlivými kvadranty realizuje jen pomocí algoritmu. Zapojení pulzního měniče je na Obr. 1.1 a). V některých aplikacích není nutné provozovat pulzní měnič ve všech čtyřech kvadrantech, ale jen ve dvou kvadrantech, tj. jedna polarita napětí a dvě polarity proudu (motorický a generátorický režim). Pro tento účel se používá zapojení podle obrázku 1.1 b) (poloviční H-můstek). Další popis bude jen pro čtyřkvadrantový pulzní měnič.



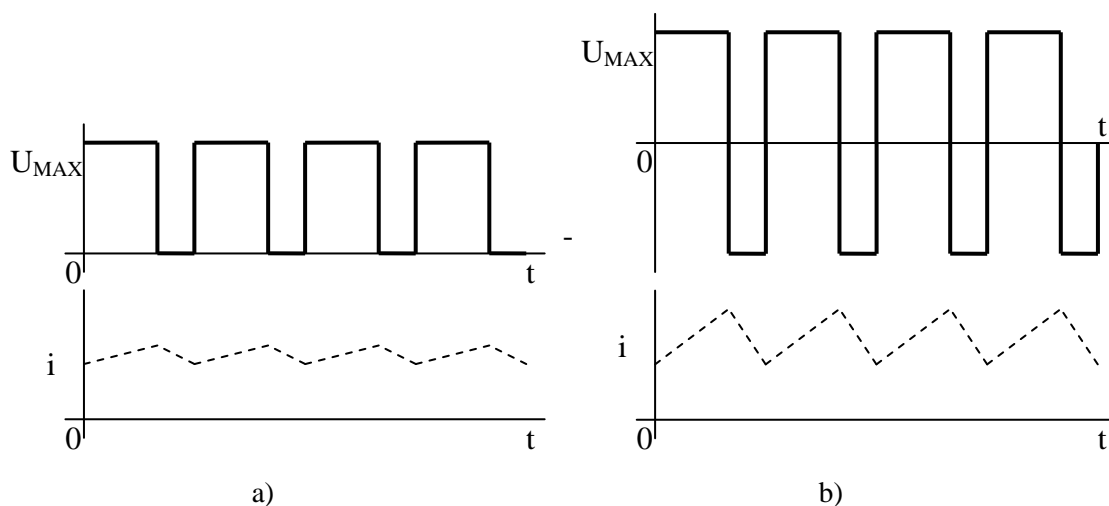
a)



b)

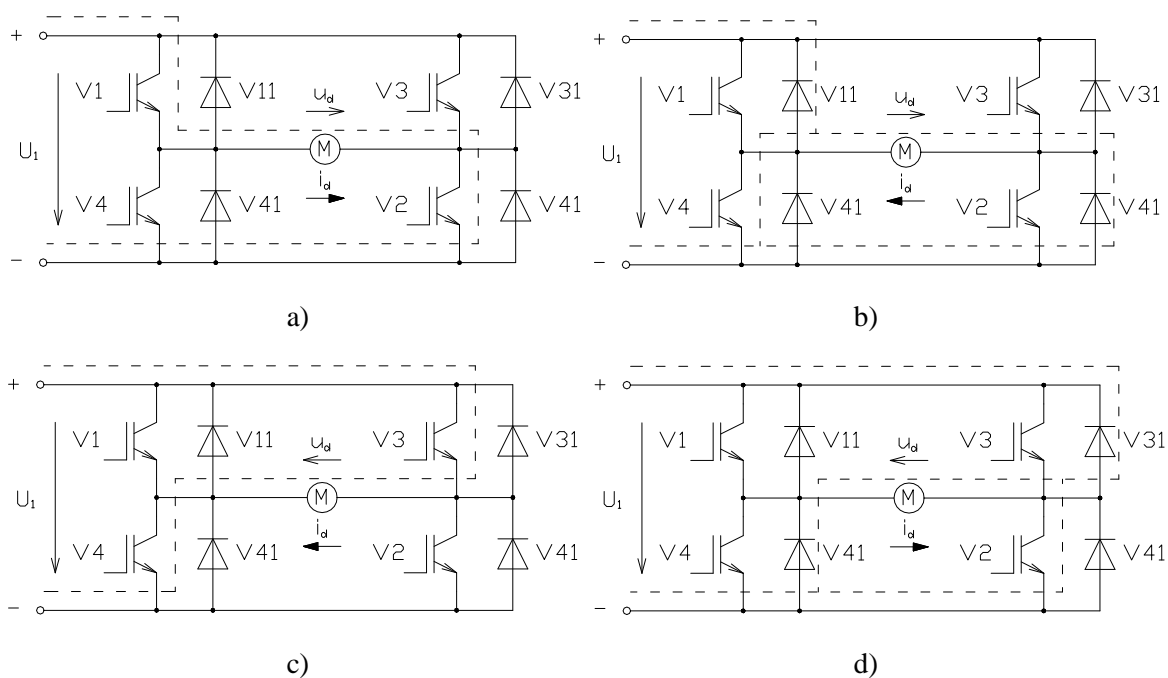
Obr. 1.1 – Pulzní měniče: a) čtyřkvadrantový, b) dvoukvadrantový

Jednotlivé prvky měniče mohou být spínány dvěma způsoby: bipolárně a unipolárně. Při bipolárním spínání se na zátěž přivádí napětí o hodnotě $\pm U_{MAX}$, u unipolárního spínání se přivádí napětí U_{MAX} pouze jedné polarity. Proud je poté v obou případech dán zatěžovatelem. U bipolárního spínání se však projevuje větší zvlnění proudu, které je dvojnásobné, oproti unipolárnímu spínání. Z tohoto důvodu bylo vybráno unipolární spínání, jež je popsáno níže. Průběhy napětí a proudu obou typů spínání je na Obr. 1.2.



Obr. 1.2 – Průběhy obou typů spínání: a) unipolární, b) bipolární

Kvadrant, ve kterém měnič pracuje, je určen prvky, které jsou v daném režimu v činnosti. Pro kladný směr napětí a proudu je buď trvale sepnut tranzistor V2 a pulzní řízení se uskutečňuje pomocí tranzistoru V1, nebo je trvale sepnut V1 a řízení se uskutečňuje tranzistorem V2. Diody V41, případně V31 (řízení pomocí V2), plní funkci nulové diody. Pro záporný směr proudu a kladné napětí (režim generátorického brzdění) se využívá řízení tranzistoru V3 nebo V4. Při jeho sepnutí je zátěž měničem zkratována přes tento tranzistor a diodu V11 nebo V21 při řízení pomocí V4. Po vypnutí tohoto tranzistoru se proud uzavírá přes diody V11 a V21 do zdroje.



Obr. 1.3 – Činnost pulzního měniče v jednotlivých kvadrantech: režimy a), c) motorický, b), d) generátorický

Při opačné polaritě napětí zátěže se v motorickém režimu buď trvale sepne V4 a řízení se uskutečňuje spínáním V3, nebo se trvale sepne V3 a řídí se pomocí V4. Funkci nulové diody přebírá V21 nebo V11 v případě trvalého sepnutí V3. V generátorickém režimu se při záporné polaritě napětí řídí tranzistor V2 nebo V1. Při vypnutém tranzistoru se v tomto případě uzavírá zpětný proud do zdroje přes V31 a V41. Činnost spínání prvků je vidět na Obr. 1.3. [1]

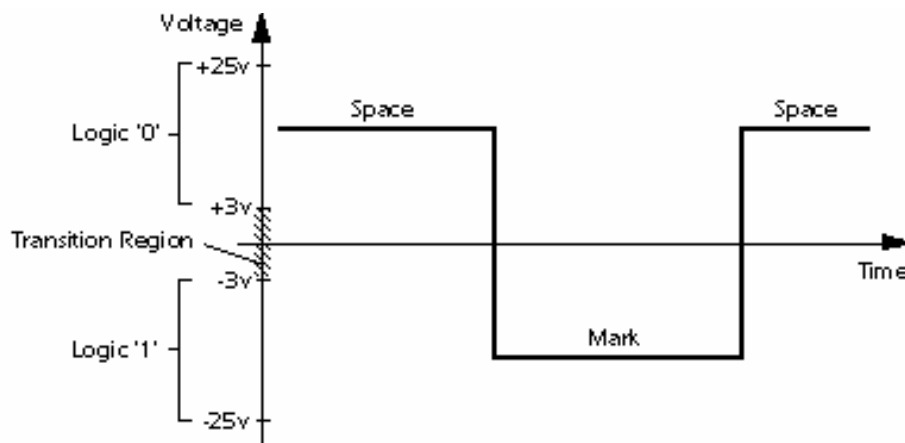
1.2 Sériová komunikace

Jedním z požadavků je, aby řídicí jednotka bylo schopná komunikovat s počítačem po sériové lince. Pro tyto účely je možné vybírat mezi několika druhy sběrnic. Lze aplikovat například tyto typy sběrnic: RS485, RS232 a USB.

Sběrnice na bázi RS485 je velice odolná proti rušení a velmi se hodí k průmyslovým aplikacím. Nevýhodou je to, že počítač není vybaven tímto typem sběrnice. Bylo by tedy nutné zkonstruovat převodník mezi RS485 a některou jinou, kterou je počítač vybaven (např. RS232). Ostatními dvěma typy sběrnic dnešní počítače disponují. Lze tedy propojit řídicí jednotku a počítač bez jakéhokoliv převodníku. Pro jednoduchost řešení byla pro řídicí jednotku zvolena sběrnice typu RS232.

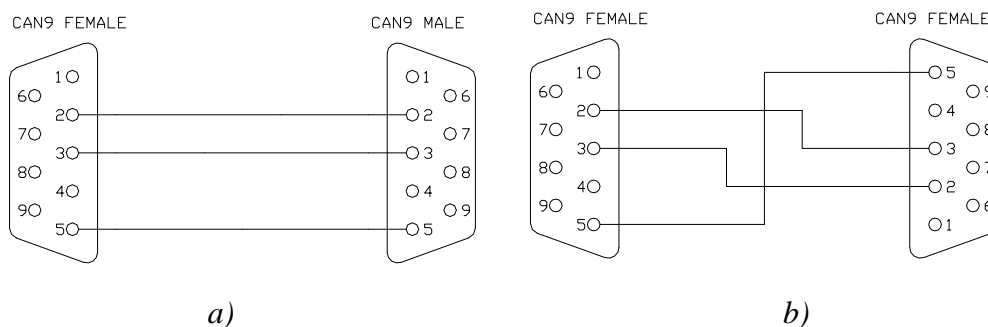
Sériová linka na bázi sběrnice RS232 je určena pro komunikaci dvou zařízení do vzdálenosti 15 m pro přenosovou rychlost 19200 Bd (snížením přenosové rychlosti lze prodloužit vzdálenost mezi zařízeními). Pro větší odolnost proti rušení je informace po propojovacích vodičích přenášena větším napětím, než je standardních 5 V. Přenos informací probíhá asynchronně, pomocí pevně nastavené přenosové rychlosti a synchronizace sestupnou hranou startovacího impulsu.

Sběrnice RS232 používá dvě napěťové úrovně (typicky ± 12 V), logickou 1 (označována jako klidový stav) a 0. Logická 1 (Mark) je indikována zápornou úrovní, zatímco logická 0 (Space) je přenášena kladnou úrovní výstupních vodičů (Obr. 1.4).



Obr. 1.4 – Napěťové úrovně RS232

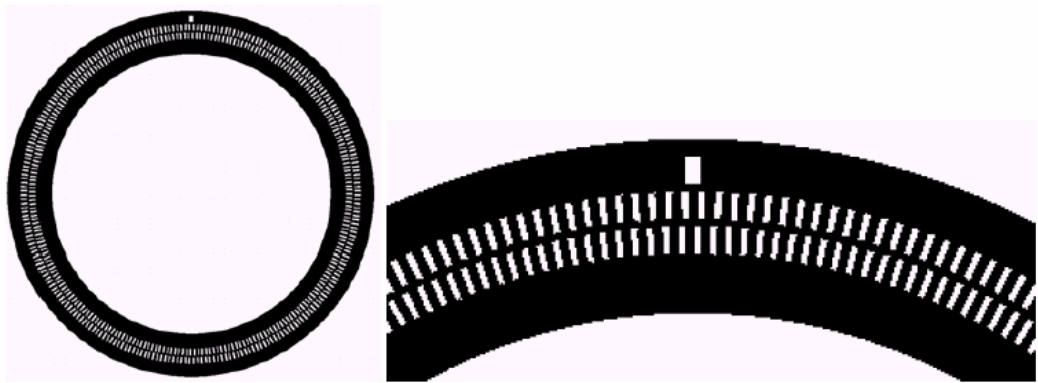
Linka RS232 je v naší aplikaci tvořena třemi vodiči (RXD – příjem dat, TXD – vysílání dat, GND – společná zem). Kompletní zapojení plně osazeného konektoru pro sériovou linku na bázi sběrnice RS232 lze nalézt na internetu. Typická zapojení propojovacích kabelů je na Obr. 1.5. Běžnější je použití kříženého kabelu. Pro druhý typ musí být křížení provedeno v připojovaném zařízení. [2]



Obr. 1.5 – 3-vodičové propojení: a) nekřížené, b) křížené

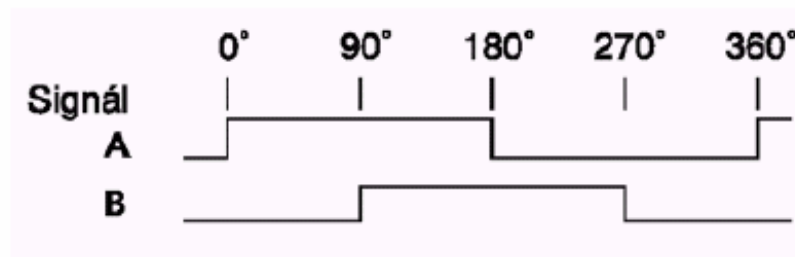
1.3 IRC čidlo

V naší aplikaci je použit inkrementální snímač otáček od firmy LARM a.s. Princip inkrementálních senzorů je založen na přerušování světelného paprsku mezi zdrojem a snímačem (např. LED a fotodioda). Výhodou je velká rozlišovací schopnost. Nevýhodou bývá relativní měření a tedy nutnost inicializace výchozí polohy. V naší aplikaci však není nutné určovat polohu.



Obr. 1.6 – Vzor snímacího kotouče

Na Obr. 1.6 je zobrazen příklad kódového kotouče optického IRC snímače a jeho detail. Optické značky na obvodu kotouče tvoří vzor průhledných a neprůhledných plošek, které při otáčení kotouče přerušují světelné paprsky snímané dvěma fotodiodami. Napětí diod je zpracováno komparátorem a signál je dále tvarován jednoduchou TTL logikou pro zlepšení strmosti hran. Konfigurace optických značek a snímacích fotodiod je taková, že výsledné signály jsou vzájemně posunuty o čtvrtinu periody impulsního signálu (Obr. 1.7). Ze vzájemného sledu náběžných hran výstupních signálů A a B lze rozlišit směr otáčení.



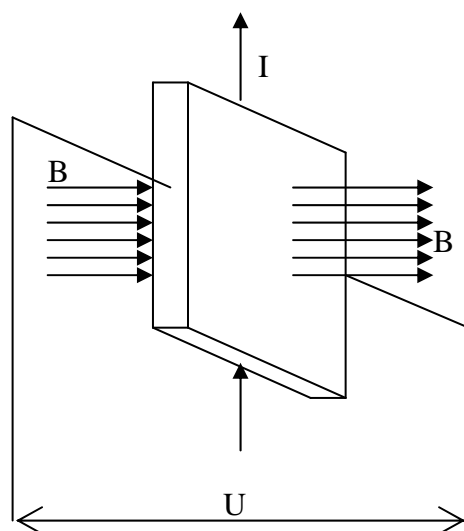
Obr. 1.7 – Průběhy signálů z IRC čidla

Mezi základní problémy IRC čidel patří elektromagnetické rušení elektronických obvodů čidla. Při tomto rušení mohou vznikat falešné pulsy na výstupech čidla, které ovlivňují správnost přijímaného signálu, čímž dochází k chybě. [3]

1.4 Měření proudu a napětí

Pro měření stejnosměrného napětí a proudu byla zvolena čidla firmy LEM. Tyto senzory využívají pro měření princip Hallova jevu. Jejich výhodou je galvanické oddělení výstupního signálu od měřeného vstupního (silového) obvodu. Není tedy potřeba následně aplikovat členy pro galvanické oddělení.

Důležitým faktorem pro další používání čidel je ten, že výstupní signál z napěťového i proudového senzoru má proudový charakter (např. 25 mA) s oběma polaritami. [4]



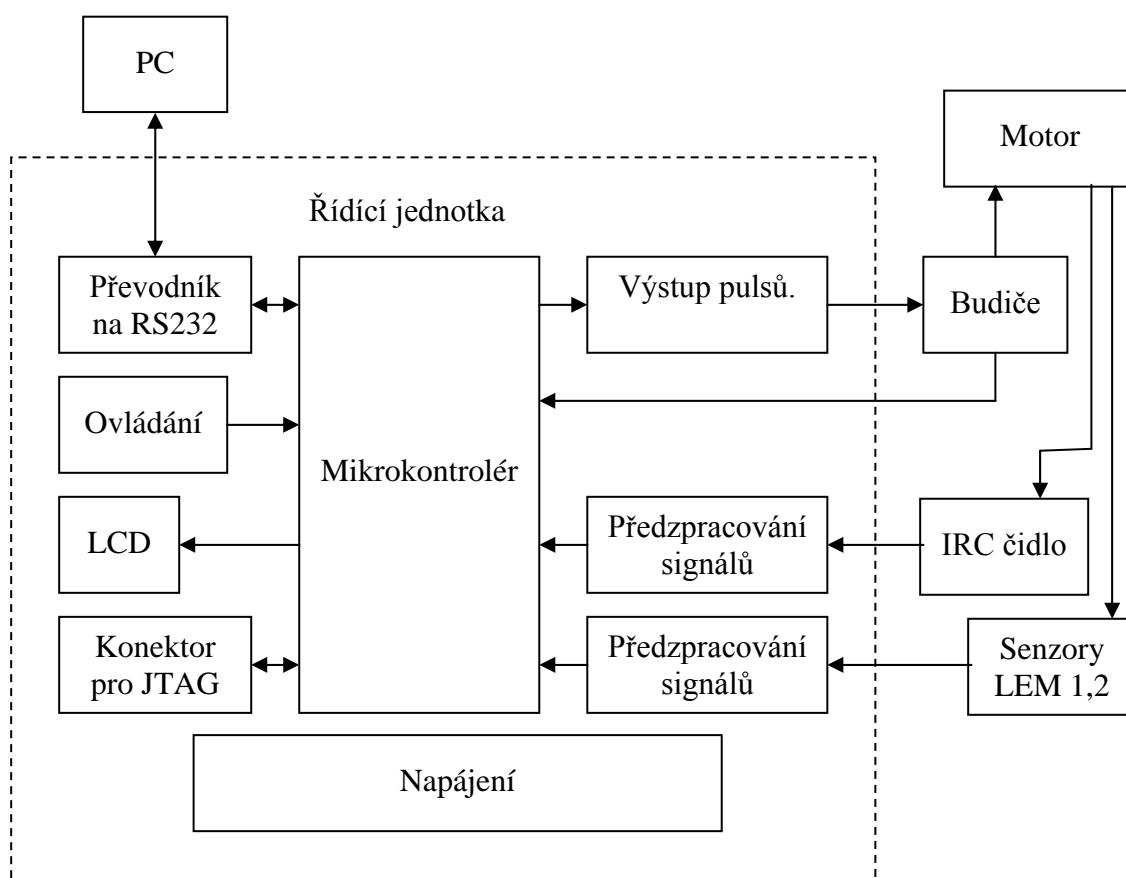
Obr. 1.8 – Hallův jev

Hallův efekt spočívá v tom, že magnetické pole kolmé na elektrický proud způsobuje zvláštní jev. Volné nosiče náboje elektrického proudu jsou v polovodičové destičce vychylovány z podélného směru a vytlačovány k okrajům. Příčinou této odchylky je elektromagnetická Lorentzova síla. Mezi okrajemi polovodiče se tak vytvoří rozdíl potenciálů, kolmý na směr proudu (Obr. 1.8).

2 VÝBĚR SOUČÁSTEK A OBVODOVÝ NÁVRH

2.1 Blokové schéma

Návrh řídicí jednotky vychází z účelu samotného použití řídicí jednotky. Při jejím návrhu je třeba znát veškeré periférie, se kterými bude řídicí jednotka spolupracovat, a samozřejmě i jejich činnost, jež je nutná ke zpracovávání příchozích nebo odchozích dat. Blokové schéma řídicí jednotky je na Obr. 2.1.



Obr. 2.1 – Blokové schéma řídicí jednotky

Základním prvkem řídicí jednotky je mikrokontrolér, ke kterému jsou připojeny ostatní prvky. Komunikace jednotky s počítačem je pomocí sériové linky na bázi sběrnice RS232. Stav řídicí jednotky a některé měřené veličiny jsou zobrazovány na modulu LCD (Liquid Crystal Display). Pro částečné řízení je vytvořen malý obslužný panel. Generované výstupní pulsy budou pomocí výstupního obvodu směřovány na příslušné spínací prvky. Dále je nutné použít obvody předzpracování signálů od veškerých senzorů, aby příchozí signál byl vhodný pro vstupní obvody mikrokontroléru.

Samozřejmostí je i napájení celé jednotky z vlastního zdroje. (Pozn.: Senzory budou taktéž napájeny z řídicí jednotky.) Pro snadné naprogramování a ladění řídicí jednotky přes rozhraní JTAG bude vyveden příslušný konektor.

2.2 Výběr součástek

2.2.1 Mikrokontrolér

Základním prvkem řídicí jednotky je mikrokontrolér. Pro řídicí jednotku je vybrán mikrokontrolér řady AVR od firmy Atmel. Než-li bude konkrétní typ mikrokontroléru vybrán, musejí být nejprve vyspecifikovány požadavky, které musí splňovat.

Mikrokontrolér musí mít dostatečně velkou paměť programu i paměť dat. Rovněž paměť EEPROM by měla být součástí. Je možné ji využít k uložení některých dat. Pro komunikaci s okolím je nutný sériový port. Při načítání analogových dat (napětí, proud, ovládání atd.) je potřeba dostatečné množství vstupů A/D převodníku. Je možné jej mít též jako externí součástku, ale většina mikrokontrolérů AVR ho má v sobě implementován. Pro ovládání čtyřkvadrantového pulzního měniče je potřeba čtyř pinů, z nichž musejí být alespoň dva schopny generovat PWM signál. K tomu je potřeba jeden 16-bitový časovač, který má minimálně dva výstupní PWM kanály. Další 16-bitový čítač bude potřeba pro zpracování signálu z IRC čidla. Další časovače budou potřeba v regulační smyčce. Neméně podstatnou záležitostí je i dostatečný počet vstupně/výstupních pinů pro další účely ovládání a načítání dat.

Z výše popsaných požadavků je nejdůležitějším parametrem počet časovačů. Dva 16bitové časovače má mikrokontrolér ATmega64. Tento typ mikrokontroléru vyhovuje všem požadavkům, v některých je dokonce převyšuje. Parametry vybraného mikrokontroléru jsou následující:

- 64 kB FLASH (paměť programu)
- 2 kB EEPROM
- 4096 B SRAM (paměť dat)
- 2x 16-bitový časovač/čítač
- 2x 8-bitový časovač/čítač
- 8x PWM kanál
- 2x sériová linka
- 8x 10-bitový A/D převodník

- rozhraní JTAG (Joint Test Action Group) pro ladění programu přímo v aplikaci
- frekvence externího krystalu 16 MHz
- napájecí napětí 4,5 – 5,5 V
- pouzdro TQFP 64

Jedná se o 8-bitový mikrokontrolér s architekturou RISC. Charakteristikou této architektury je, že obsahuje redukovanou sadu instrukcí; většina instrukcí se provede v jednom cyklu; zřetězení dat (tzv. pipelining), kde zatímco jedna instrukce je prováděna, druhá je načítána z programové paměti. Šíře instrukčního slova je 16 bitů (paměť programu je organizována po slovech, což vede k zvětšení požadavků na paměť). Instrukce jsou 16 a 32 bitové. Všechny tyto charakteristiky vedou k vyšší rychlosti.

Mikrokontroléry využívají koncepci Harwardské architektury – oddělení paměti programu a dat. Jádro mikrokontroléru se skládá z ALU a ze 32 stejných 8-bitových pracovních registrů. Posledních šest registrů můžeme ve dvojici použít jako ukazatele pro nepřímé adresování paměti. Paměťový prostor dále obsahuje adresy vstupně/výstupních registrů sloužících k periferním funkcím.

Strojový cyklus trvá jeden takt oscilátoru, není použita totiž žádná předdělička. Při použití 16 MHz krystalu tak lze dosáhnout výpočetního výkonu až 16 MIPS (využito výhod pipelingu).

Paměť programu je „In-Systém Programmable Flash Memory“. To znamená, že kromě klasického paralelního programování této paměti, je možné sériové programování přímo v aplikaci (rozhraní ISP nebo JTAG). Tento typ mikrokontroléru je velice vhodný pro programování v jazyce C. [5]

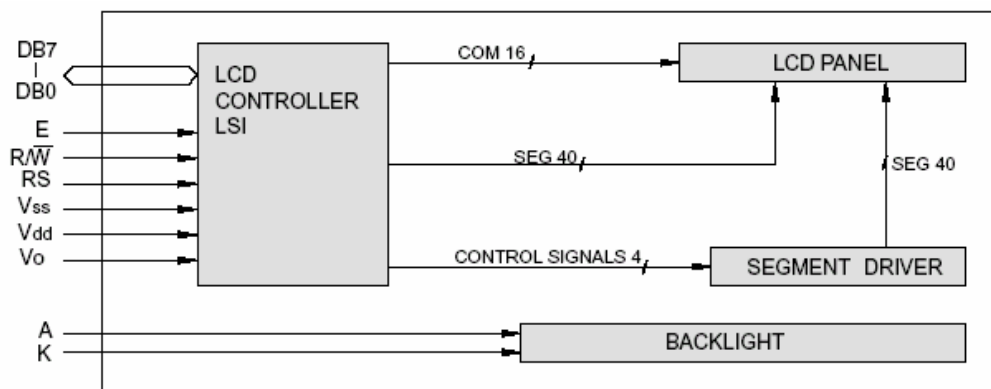
2.2.2 Displej

Pro zobrazování vykonávaných instrukcí, aktuálních hodnotách řízení atd. byl vybrán alfanumerický LCD zobrazovač typu PC1602-F. Tento displej obsahuje řadič S6A0069, který je kompatibilní se standardním řadičem LCD HD44780. Znamená to tedy, že vybraný displej má stejnou instrukční sadu a stejné ovládání. Vybraný displej má 2x16 znaků.

Rozhraní displeje tvoří datová sběrnice a tři řídicí linky (Obr. 2.2). Displej má jen dva registry: instrukční a datový, které se volí linkou RS. Jako taktovací signál je použita linka označená E. Signálem R/W se určuje zda budou data do řadiče posílána nebo z něj načítána.

Znaková sada je standardní anglická (+ obvyklé čínské znaky). Kód znaku je ASCII. Do paměti DDRAM (Display Data Random Acces Memory) jsou ukládána data (včetně pozice na displeji), jež se mají zobrazit. Do paměti CGRAM (Character Generator Random Acces Memory) se ukládají uživatelem nově vytvořené znaky (maximálně 8). Velikost znaku je 5x8 bodů.

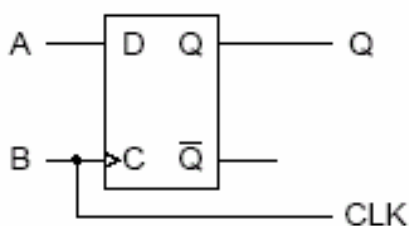
Komunikace displeje (resp. řadiče) může být osmibitová nebo čtyřbitová (data jsou vysílána nadvrát). Na počátku je vždy nutné displej inicializovat, tj. vyslat instrukci, která určí typ komunikace, počet řádků atd. Většina dnešních zobrazovačů nabízí podsvícení displeje pomocí LED. Samozřejmě je i nastavení vhodného kontrastu displeje. [6]



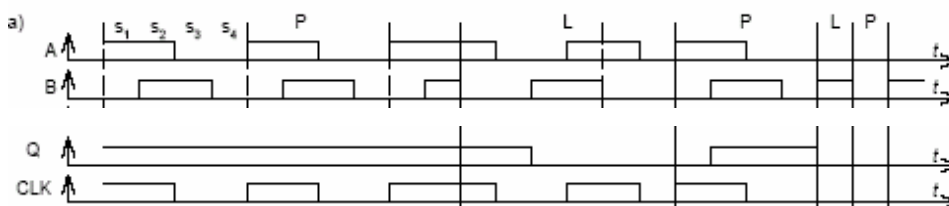
Obr. 2.2 – Blokové schéma LCD

2.2.3 Ostatní obvody

Při zpracování signálu z IRC čidla je třeba zajistit, aby se při změně signálů A a B vyhodnotil směr. To lze samozřejmě učinit softwarově v mikrokontroléru. Přináší to s sebou, však problémy. Mikrokontrolér by musel mít další 16-bitový časovač. Navíc by mohlo dojít ke zpomalení reakce řídicí jednotky. Těmto problémům se lze velice jednoduše vyhnout, použijeme-li externí zařízení. K dekodování směru lze použít jednoduchou logiku; klopný obvod typu D (4013). Na vstup D připojíme jeden ze signálů (A nebo B). Druhý připojíme na hodinový vstup (Obr. 2.3). Výstupní signál Q klopného obvodu nám udává směr otáčení . Průběh je na Obr.2.4.



Obr. 2.3 – Obvod pro převod signálů A a B na směr a CLK



Obr. 2.4 – Průběh signálů z IRC čidla a D-KO

Dalším důležitým obvodem je MAX232, který vysílá a přijímá data po sériové lince. Protože mikrokontrolér vysílá data na sériovou linku v úrovni TTL je nutné použít tento obvod pro konverzi úrovní TTL/RS232. Obvod obsahuje dvě dvojice oddělovačů konvertujících napěťové úrovně. Napětí pro RS232 se získává pomocí nábojové pumpy. Tento obvod si tak vystačí s +5 V a potřebné napětí si samostatně vyrobí pomocí čtyř externích kondenzátorů.

Signál z napěťového a proudového čidla je potřeba před zpracováním nejprve vhodně upravit. A/D převodník mikrokontroléru vyžaduje, aby na jeho vstupu byla hodnota 0 až 5 V. Proto je třeba převést bipolární signál ze sensorů na unipolární. K tomu nám poslouží dva operační zesilovače (TL072) v invertujícím zapojení se zesílením 1. Na neinvertujícím vstupu nastavíme tzv. offset, tzn. hodnotu odpovídající nulovému proudu. V našem případě to bude činit 2,5 V. Přicházející proudový signál tak musíme upravit na $\pm 2,5$ V. Výstupní signál pak bude dán rozdílem mezi vstupem a nastaveným offsetem. Polarita bude zjišťována dodatečně v softwaru. [6][7]

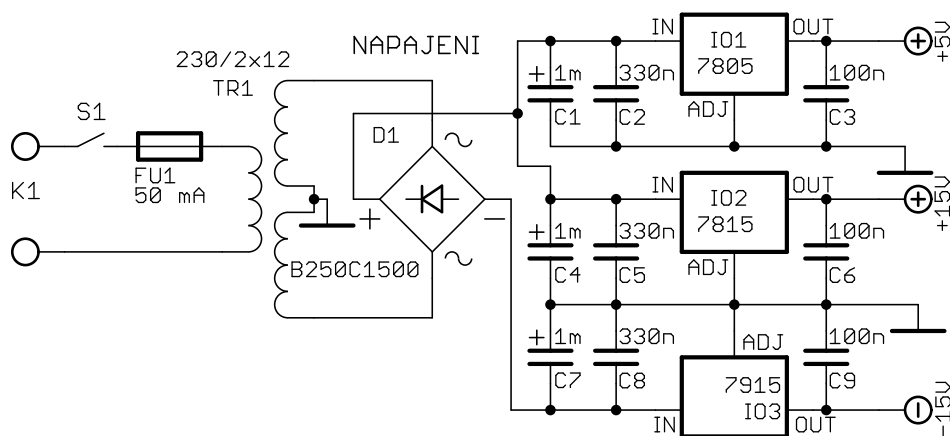
2.3 Obvodový návrh

2.3.1 Napájení

Pro správnou funkčnost řídicí jednotky je potřeba zajistit spolehlivé napájení. Řídicí jednotka bude z hlediska napájení nezávislá na jiných zdrojích (např. již snížené napětí).

Napájení mikrokontroléru a většiny ostatních integrovaných obvodů vyžaduje stabilizované napětí +5 V. Operační zesilovače však potřebují souměrné napětí ± 15 V. Kromě operačních zesilovačů toto napětí také vyžadují senzory napětí a proudu (LEMy), které budou z řídicí jednotky napájeny. Rovněž IRC čidlo bude napájeno prostřednictvím řídicí jednotky (napětím +5 V).

Základem napájecího obvodu je síťový transformátor. Jeho výběr závisí na požadavcích používaných napětí a příslušných odběrech. Je nutné vytvořit souměrný zdroj napětí. To lze provést dvěma způsoby: s jedním sekundárním vinutím nebo se dvěma sekundárními vinutími. První případ však změkčuje zdroj napětí, což by mohlo vést k nežádoucímu ovlivnění senzorů LEM. Zvolen byl tedy druhý způsob. Dalším důležitým faktorem je maximální dovolený odběr proudu jednotlivých součástí. Přibližný odběr bude činit maximálně 450 mA. Tento odběr je vztažen na jedno sekundární vinutí transformátoru. Ve výsledku bude celkový proud dvojnásobný, tj. 900 mA. Pro vytvoření souměrného napětí bude postačovat, aby výstupní napětí transformátoru mělo hodnotu 12 V (po usměrnění a vyfiltrování získáme přibližně hodnotu 17 V). Vybraný transformátor tedy má výkon 12 VA. Schéma zapojení je na Obr 2.5.

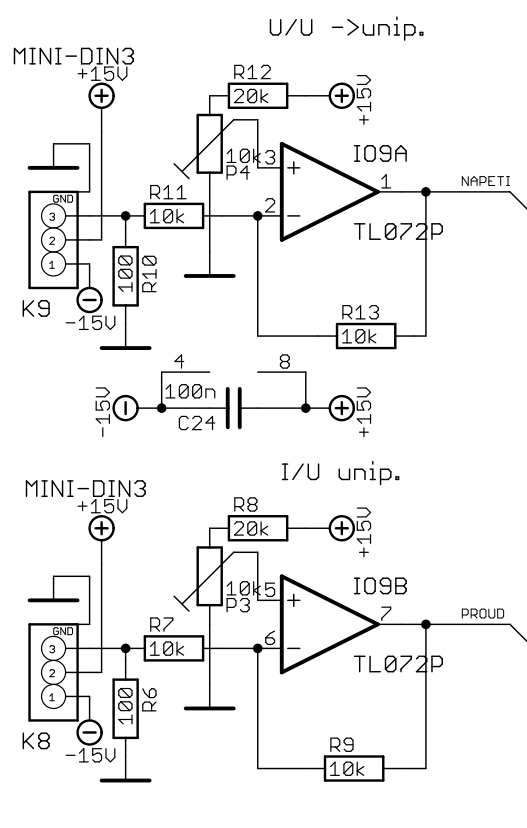


Obr. 2.5 – Schéma obvodu napájení

Přívodní síťové napětí je přes konektor K1 (KES 2), síťový vypínač S1 (P-M101) a trubičkovou pojistku FU1 (50 mA) přivedeno na primární vinutí transformátoru TR1 (230 V / 2x12 V). Sekundární vinutí jsou jedním koncem spojena se zemí. Zbylé dva vývody jsou připojeny na integrovaný diodový můstek D1 (B250C1500).

Pro napětí +5 V je použit stabilizátor IO1 (7805) s chladičem. Pro souměrné napětí jsou vybrány stabilizátory IO2 a IO3 (7815 – výstupní napětí +15 V, resp. 7915 – výstupní napětí -15 V) bez chladičů. Ke každému stabilizátoru jsou připojeny blokovací kondenzátory (C2, C3, C5, C6, C8, C9) s hodnotami doporučenými výrobcem (330 nF a 100 nF). Před každým stabilizátorem je umístěn kondenzátor (C1, C4, C7) o velikosti 1 mF, který filtruje napětí z usměrňovače.

2.3.2 Zpracování proudu a napětí



Obr. 2.6 – Schéma obvodu zpracování signálů z LEMů

Jak bylo uvedeno výše bipolární signál z napěťového a proudového senzoru je třeba upravit na hodnotu 0 až 5 V pro vstupní obvody A/D převodníku mikrokontroléru. Základen jsou dva operační zesilovače IO9A a IO9B (TL072; 2x operační zesilovač v pouzdru DIL8), ke kterým je připojen blokovací kondenzátor C24 (100 nF). Oba zesilovače jsou v invertujícím zapojení napájeny ze souměrného zdroje ± 15 V.

Důležitou podmínkou pro správné zpracování signálu je nastavení offsetu. Na neinvertující vstup je připojen napěťový dělič napájený +15 V. Protože hodnota offsetu (+2,5 V) musí být velmi přesná, je nutné dělič správně nastavit. Proto je dělič tvořen rezistorem R8 (resp. R12) o velikosti 20 k Ω a vysokootáčkovým trimrem P3 (resp. P4) o velikosti 10 k Ω , kterým nastavíme přesně požadovanou hodnotu.

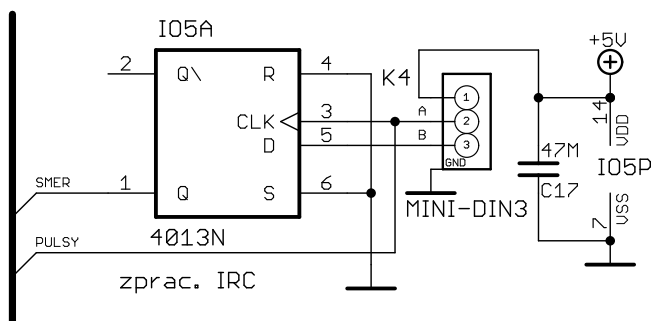
Signál ze sensorů LEM má proudový charakter (předpoklad ± 25 mA). Dle doporučení výrobce je k senzoru paralelně připojen rezistor R6 (resp. R10) o velikosti 100 Ω . Takto získáme napěťový signál o velikosti $\pm 2,5$ V, což je velmi výhodné, protože tento signál můžeme přivést přes rezistor R7 (resp. R11; 10 k Ω) na invertující vstup.

Hodnota rezistoru ve zpětné vazbě je ovlivněna potřebným zesílením. Jelikož však výstupní signál z operačního zesilovače má již požadovanou napěťovou úroveň, není potřeba tento signál zesilovat (zesílení $A = 1$). Hodnoty rezistorů R9 (resp. R13) ve zpětné vazbě jsou tedy 10 k Ω .

Signály od sensorů jsou přivedeny na konektory K7 a K8 (MINI-DIN3), přes které jsou senzory rovněž napájeny. Schéma zapojení je na Obr. 2.6.

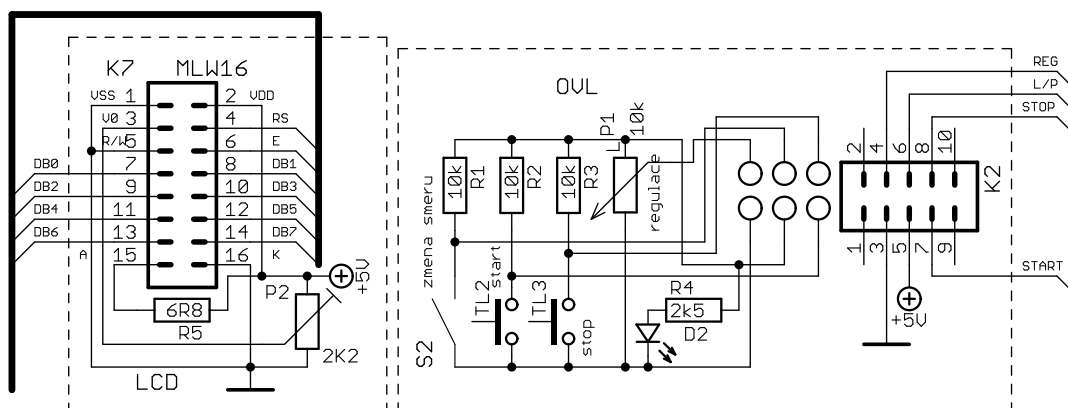
2.3.3 Zpracování signálu z IRC čidla

Obvod (Obr. 2.7) pro zpracování signálu z IRC čidla je velmi jednoduchý. Jak již bylo zmíněno veškerou činnost obstará jeden D klopný obvod. Na schématu je to obvod IO5 (4013; 2x D-klopný obvod v pouzdře DIL14) s blokovacím kondenzátorem C17 (47 μ F). Signály A a B jsou přivedeny na konektor K4 (MINI-DIN3), přes který je rovněž zajištěno napájení senzoru. Výstupem tohoto zapojení jsou opět dva signály, jeden nám určuje směr otáčení a druhý dává počet pulsů na otáčku



Obr. 2.7 – Schéma obvodu zpracování signálů z IRC čidla

2.3.4 LCD panel a ovládání



Obr. 2.8 – Schéma obvodů zapojení displeje (vlevo) a ovládacího modulu (vpravo)

Jelikož má mikrokontrolér dostatek vývodů, bylo zvolena 8-bitová komunikace s displejem. Použitý displej PC1602-F je spojen se základní deskou přes konektor K7 (MLW16). Protože se nepředpokládá čtení dat z displeje, je ovládací vstup RW uzemněn. Ostatní řídicí a datové vstupy jsou připojeny k portům mikrokontroléru. Schéma zapojení je na Obr. 2.8.

K řízení kontrastu displeje slouží trimr P2 (2,2 k Ω). Pro podsvětlení displeje je potřeba snížené napětí 4,1 V, které se dosáhne předřadným rezistorem R5 (6,8 Ω). Podsvětlení lze případně vypnout pomocí jumperu.

Ovládací panel je spojen se základní deskou konektorem K2 (MLW10). Na ovládacím panelu je umístěna signalizace napájení pomocí LED (D2) s předřadným rezistorem R4 (2,5 k Ω). Tlačítka TL2 a TL3 (P-DT6) jsou zapojena standardně tak, aby jejich sepnutí bylo signalizováno úrovní log. 0. Obdobně je to zapojeno u dvupolohového přepínače polarity S2 (MTS 500APC). Klidový stav (úroveň log. 1) je definován vyvažovacími rezistory R1 až R3 (10 k Ω). Pro regulaci otáček je použit potenciometr P1 (10 k Ω).

2.3.5 Obvody mikrokontroléru a další obvody

Mikrokontrolér (IO7) je připojen ke stabilizovanému napětí +5 V. Blokování napájení je provedeno pomocí kondenzátorů C14 a C16 (100 nF). a skupinových kondenzátorů C13 a C15 (1 μ F). Jelikož v aplikaci je využito A/D převodníku, je na pin AVCC a AREF přivedeno napětí +5 V a na pin AGND zem.

K pinům XTAL1 a XTAL2 je připojen krystal Q1 o frekvenci 16 MHz, který je na obou koncích spojen přes kondenzátory C11 a C12 (22 pF) se zemí.

Pro zajištění resetu mikrokontroléru je použit obvod DS1233 (IO6), který je vybaven monitorováním napájení (napěťový watchdog). Reset je aktivní v úrovni log. 0. Tlačítko TL1 (P-DT6) s paralelně připojeným kondenzátorem C10 (500 pF) slouží k ručnímu resetu.

Kompletně celý port A je využit pro datovou komunikaci s displejem (označení DB0 až DB7). Dva řídicí signály (E, RS) se nacházejí na pinech PC7 a PC6 portu C.

Čtyři piny portu B jsou využity ke generování pulsů. Na pinech PB5 a PB6 se nacházejí výstupy komparační jednotky 16-bitového časovače T1, které slouží pro tvorbu PWM signálů. Piny PB7 a PB4 jsou použity jako výstupní, a slouží k otevírání/zavírání příslušných tranzistorů můstku. Tyto čtyři signály jsou přivedeny na čtyři hradla AND obvodu IO8 (4081; 4x AND v pouzdře DIL14), s blokovacím kondenzátorem C23 (47 μ F). Výstup každého hradla je směřován na příslušný tranzistor můstku. Podle užitého algoritmu jsou otevírána příslušná hradla a PWM signál se tak dostává na dva tranzistory (podrobněji v kapitole 6). Zároveň tato hradla slouží pro zablokování signálů z řídicí jednotky. Pin PDO pracuje jako vstupní a snímá chybovou informaci od budičů. Všechny tyto signály jsou spojeny s budičem přes konektor K6 (CAN9-F).

Na portu E se nachází většina řídicích signálů. Piny PE7 a PE6 načítají informace z IRC čidla (resp. D-klopného obvodu). Pin PE6 (zachytávací jednotka 16-bitového čítače/časovače T3) načítá pulsy, z pinu PE7 se získává směr otáčení. Piny PE4, PE3, PE2 slouží k načítání informací z ovládacího panelu (start, stop, polarita).

Sériová linka k použití pro sběrnici RS232 je vyvedena z pinů PE1 a PE0 (USART0) na obvod MAX232 (IO4, pouzdro DIL16). Využit je pouze jeden přijímač a jeden vysílač obvodu. Jako nábojové pumpy jsou použity kondenzátory C19 až C22 (1 μ F, dle doporučení výrobce). Blokovací kondenzátor C18 má hodnotu 100 nF. Sériová linka je vyvedena na konektor K5 (CAN9-F). Protože jednotka bude propojována s počítačem pomocí nekříženého kabelu, jsou piny 2 a 3 překříženy již na desce.

Na portu F se nachází 8 pinů A/D převodníku. Piny PF4 až PF7 jsou však použity pro programování mikrokontroléru přes rozhraní JTAG, které je vyvedeno na standardní konektor MLW10 (K3). (Pozn. toto rozhraní neumožňuje další využití těchto pinů.) Piny PF2 a PF3 konvertují signál z čidel napětí a proudu do digitální podoby. Na pin PF1 je přiveden signál od potenciometru P1, který slouží k zadávání velikosti požadované hodnoty regulované veličiny.

Všechny ostatní nevyužité piny jsou vyvedeny na lámací lišty pro případ, kdyby bylo nutné některý z nich ještě použít. Celkové schéma zapojení řídicí jednotky je uvedeno v příloze.

2.4 Provedení řídicí jednotky

Pro návrh desek plošných spojů byl použit program EAGLE 4.16. Rozmístění součástek bylo voleno tak, aby umožnilo snadnou montáž, přístup ke konektorům a tlačítkům. Všechny součástky řídicí jednotky jsou rozmístěny na třech deskách. Jedná se desku ovládacích prvků, desku mikrokontroléru a základní desku (ostatní součástky a obvody). Při návrhu byly použity z větší části součástky s drátovými vývody.

Všechny integrované obvody jsou uloženy v patičkách. Jelikož pouzdro mikrokontroléru (TQFP64) je určeno pro povrchovou montáž, je mikrokontrolér a jeho nejbližší součástky (krystal a nejbližší kondenzátory) provedeny technologií SMD (Surface Mount Device) na samostatné desce. Propojení se základní deskou je pomocí konektorů na okrajích desky (lámací a dutinkové lišty).

Řídicí jednotka je umístěna v plastové krabičce KMP60 (160x140x60 mm). Základní deska s deskou mikrokontroléru se nachází ve spodní části krabičky. Na základní desce se nachází resetovací tlačítko. Pro účely ladění programu jsou na lámací lištu vyvedena napětí +5 V, +15 V a samozřejmě GND. Konektor pro rozhraní JTAG se nachází rovněž uvnitř krabičky.

Zbývající konektory jsou taktéž umístěny na základní desce, ale přístup k nim je přes přední panel krabičky. Na zadním panelu se nachází síťový konektor K1 a síťový vypínač S1.

Výslednou podobu řídicí jednotky lze vidět na Obr. 2.9.



Obr. 2.9 – Provedení řídicí jednotky

3 NASTAVENÍ REGISTRŮ MIKROKONTROLÉRU

Velkými výhodami všech mikrokontrolérů řady AVR ATmega jsou jejich periférie, které umožňují vhodně zpracovávat vstupní i výstupní data. Mikrokontrolér Atmega64 je velice dobře vybaven a to zejména v oblasti časovačů. V následujících řádcích budou popsány vlastnosti periférií a nastavení příslušných registrů. Zejména se jedná o periférie týkající se časovačů a sériového portu. Popsány jsou pouze využití funkce a jejich vlastní nastavení. Další periférie budou popsány stručněji. Více se lze dozvědět na stránkách výrobce [5].

3.1 Časovače/čítače

Mikrokontrolér Atmega64 je vybaven dvěma 8-bitovými a dvěma 16-bitovými časovači/čítači. 8-bitové časovače/čítače mikrokontroléru jsou označeny jako Timer/Counter0 a Timer/Counter2, 16-bitové jako Timer/Counter1 a Timer/Counter3. Ve výsledném programu nebyl použit pouze Timer/Counter2.

Časovače/čítače nabízejí volitelnou předděličku. Její hodnota je např. 1, 8, 32, 64, 128, 256 atd., dle typu časovače/čítače. Kromě základních funkcí časovače nebo čítače lze využít další funkce: výstupní komparační jednotka pro generování obdélníkového průběhu, záchytná jednotka a generování různých režimů PWM signálu.

3.1.1 Timer/Counter0

Tento 8-bitový časovač/čítač je použit ve funkci časovače (pro cyklus regulační smyčky). Následující 8-bitové registry jsou pro obsluhu časovače: TCCR0 (Timer/Counter Control Register) je řídicí registr, TCNT0 (Timer/Counter Register) je vlastní časovač. Obsluha přerušení je v registrech TIFR (Timer/Counter Interrupt Flag Register) a TIMSK (Timer/Counter Interrupt Mask Register).

Časovač pracuje v tzv. „Normálním módu“. Směr čítání je pouze nahoru, při každém taktu hodin časovače se hodnota v registru TCNT0 zvýší o jedničku. Po dosažení hodnoty 255 časovač přeteče do 0 a nastaví se příznak TOV0 (Timer/Counter Overflow Flag) v registru TIFR. Povolením přerušení v registru TIMSK dojde k jeho vygenerování. Nulování příznaku přerušení dochází v rutíně obsluhy přerušení.

Pokud nevyhovuje perioda přetečení časovače, lze časovač softwarově přednastavit zápisem do registru TCNT0. Požadovaná hodnota se vypočte následujícím způsobem:

$$TCNT0 = 2^R - \frac{F_{OSC} \cdot t_{OV}}{n}, \text{ kde} \quad (2)$$

R – počet bitů časovače (8)

F_{OSC} – frekvence oscilátoru (16 MHz)

t_{OV} – žádaná perioda přetečení (2 ms)

n – předdělička časovače (1024)

Pro požadované přerušení po 2 ms je hodnota nastavení registru TCNT0 0xE1.

Přehled nastavení jednotlivých bitů registrů pro obsluhu Timer/Counter0 je níže.

TCCR0 – řídicí registr

bit	7	6	5	4	3	2	1	0
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00
set	0	0	0	0	0	1	1	1

- FOC0 – vynucení signálu shody komparace (nevyužito)
- WGM00 a WGM01 – výběr módu časovače (normální mód)
- COM00 a COM01 – výběr chování pinu OC0 (nevyužito)
- CS00, CS01 a CS02 – výběr zdroje taktovacího kmitočtu a předděličky. Zároveň se nastavením těchto bitů zapíná (resp. vypíná) časovač. (předdělička 1024)

TIMSK – povolení/zákaz přerušení

bit	7	6	5	4	3	2	1	0
	ostatní časovače						OCIE0	TOIE0
set							0	1

- OCIE0 – povolení/zákaz přerušení při komparaci (nevyužito)
- TOIE0 – povolení/zákaz při přetečení časovače

TIFR – příznaky přerušení

bit	7	6	5	4	3	2	1	0
	ostatní časovače						OCF0	TOV0
read							0/1	0/1

- OCF0 – příznak dosažení shody při komparaci (nevyužito)
- TOV0 – příznak přetečení časovače

3.1.2 Timer/Counter1

Následující 16-bitový časovač/čítač má úlohu generování PWM signálu. Řídící registry jsou 8-bitové: TCCR1A, TCCR1B a TCCR1C (Timer/Counter Control Register A, B, C). Timer/Counter1 obsahuje tři komparační jednotky OC1A, OC1B a OC1C (Output Compare Register), využity jsou pouze první dvě. Spolu s registrem TCNT1 (Timer/Counter Register), časovací registr, se jedná o 16-bitové registry, které jsou rozděleny na dva 8-bitové (s doplňujícím označením H nebo L, např. TCNT1L). Přerušení není v tomto případě použito generováno.

Generování PWM signálu je nabízeno v několika různých módech. Zvolen byl mód „Fast PWM“. Lze měnit frekvenci a zároveň rozlišení PWM nastavením TOP hodnoty. Rozlišení bylo zvoleno 10-bitové.(TOP hodnota je tedy 1023). Frekvence PWM se vypočte takto:

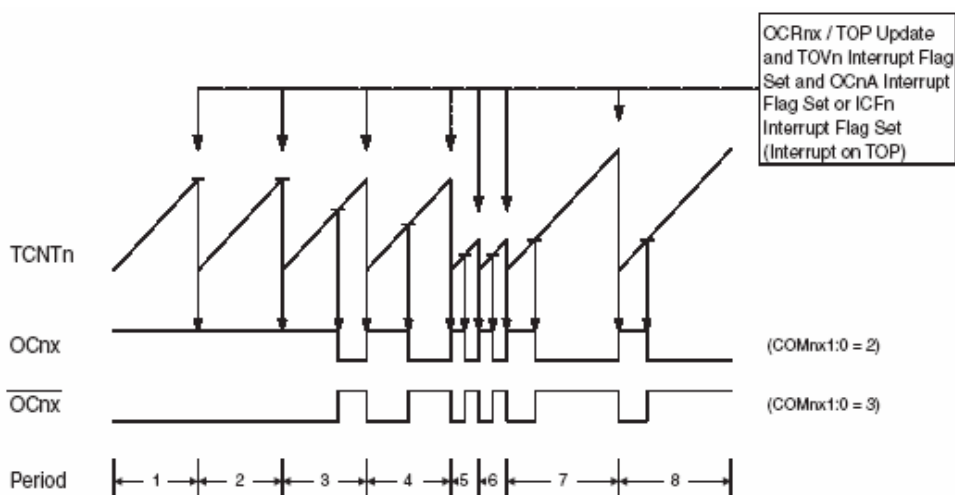
$$f_{PWM} = \frac{F_{OSC}}{n \cdot (1 + TOP)}, \text{ kde} \quad (3)$$

F_{OSC} – frekvence oscilátoru (16 MHz)

n – předdělička časovače (8)

Frekvence PWM poté vychází 2kHz. Zápisem 0x01FF do registru TCNT1 je nastavena TOP hodnota a samozřejmě rozlišení.

Hodnotou v registrech OCR1A a OCR1B lze nastavit střihu PWM. Obsah těchto registrů je porovnáván s obsahem TCNT1. V případě shody bude pin OC1A (resp. OC1B) nastaven do 0, při přetečení se piny nastaví do 1 (tzv. neinvertovaná PWM). Lze volit mezi neinvertovanou a invertovanou PWM (viz Obr. 3.1). Aby se signál fyzicky objevil musí být piny OC1A a OC1B nastaveny jako výstupní.



Obr. 3.1 – Fast PWM

Jestliže jsou registry OCR1A nebo OCR1B rovny TOP hodnotě, nebude se výstup měnit. Když se však tyto registry rovnají 0, budou se na výstupu objevovat úzké špičky (nelze docílit střídý 0%), jejichž délka bude shodná s délkou jednoho taktu časovače.

Přehled nastavení jednotlivých bitů registrů pro obsluhu Timer/Counter0 je níže.

TCCR1A – řídicí registr A

bit	7	6	5	4	3	2	1	0
	COM1A1	COM1A0	COM1B1	COM1B0	COM1C1	COM1C0	WGM11	WGM10
set	1	0	1	0	0	0	1	1

- COM1A(B,C)0 a COM1A(B,C)1 – výběr chování pinů OC1A (B,C). Použity pouze komparační jednotky A a B. (neinvertující mód)

TCCR1B – řídicí registr B

bit	7	6	5	4	3	2	1	0
	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10
set	0	0	0	0	1	0	1	0

- WGM10 až WGM13 – výběr módu časovače (Fast PWM, 10-bitová)
- CS00, CS01 a CS02 – výběr zdroje taktovacího kmitočtu a předděličky. (předdělička 64)

TCCR1C – řídicí registr C

bit	7	6	5	4	3	2	1	0
	FOC1A	FOC1B	FOC1C	-	-	-	-	-
set	0	0	0	0	0	0	0	0

- FOC1A, FOC1B, FOC1C – vynucení signálu shody komparace (nevyužito)

3.1.3 Timer/Counter3

Timer/Counter3 má naprosto stejné vlastnosti jako předcházející Timer/Counter1. Názvy registrů jsou stejné, liší se pouze číselným označením. Tento časovač/čítač opět pracuje v „Normálním módu“, ale tentokrát jako čítač (pro načítání pulsů z IRC). Vstupní signál je přiveden na pin T3 (nastaven jako vstupní). Podle výrobce mikrokontroléru může být maximální frekvence vstupního signálu maximálně 2,5krát nižší, než-li frekvence oscilátoru. Tato podmínka je v našem případě splněna, protože frekvence pulsů z IRC, nebude tak vysoká. Zvýšení hodnoty o 1 v registru TCNT3 může být vyvoláno jak vzestupnou tak sestupnou hranou (v tomto případě na tom nezáleží).

Jak bylo zmíněno, podoba a význam registrů (včetně jednotlivých bitů) je totožný s Timer/Counter1. Proto zde bude znázorněno jen nastavení registrů a to takto: TCCR3A = 0x00, TCCR3B = 0x07, TCCR3C = 0x00.

3.2 Sériový port

Ke komunikaci po sériové lince slouží sériový port. Mikrokontrolér ATmega64 je vybaven dvěma porty, z nichž je využit pouze jeden s označením USART0 (Universal Synchronous and Asynchronous serial Receiver and Transmitter). Jedná se o duplexní sériový kanál, umožňující asynchronní nebo synchronní komunikaci. Formát přenášených dat je variabilní, rozsah 5 až 9 bitů, 1 nebo 2 stop bity, parita žádná, lichá nebo sudá. Sériový port je schopen detekovat falešný start bit, chybný znak, přetečení přijímacího bufferu (zásobníku), chybnou paritu atd.

Pro obsluhu jsou určeny řídicí registry UCSR0A, USCR0B a UCSR0C (USART Control and Status Register) a datový registr UDR0 (USART Data Register). Datový registr je určen jak pro vysílání tak pro příjem. Ve skutečnosti se jedná o dvě fyzicky oddělená místa. Zápisem do UDR0 se zapíše data do vysílacího registru, čtením z UDR0 se data čtou z přijímacího registru.

Piny pro sériový port mají označení TxD (vysílání) a RxD (příjem), pin XCK není použit (jen pro synchronní přenos). Vysílač je jednoduše bufferován, protože se skládá z datového registru UDR0 a posuvného vysílacího registru. Tím je možné vysílat bez přestávky mezi jednotlivými rámci. Přijímač se skládá rovněž z posuvného registru, datového registru UDR0 a další logiky, která kontroluje konzistenci rámce, správné časování atd.

Logika pro generování hodin sériového portu obsahuje čítač, jenž slouží pro generování taktovacího signálu a jehož obsah se každým taktem oscilátoru sníží o 1. Přednastavením tohoto čítače se určuje přenosová rychlost sériové linky. Pro tuto předvolbu slouží registr UBRR0 (USART Baud Rate Register). Jedná se o 16-bitový registr, který je rozdělen na dva 8-bitové (obdobné značení jako u časovačů). Hodnotu pro nastavení tohoto registru lze získat z následujícího vzorce:

$$UBRR0 = \frac{F_{osc}}{16 \cdot BAUD} - 1, \text{ kde} \quad (4)$$

F_{osc} – frekvence oscilátoru (16 MHz)

BAUD – přenosová rychlost (38400 Bd)

Hodnota pro registr UBRR0 je potom 0x19. Vypočtením této hodnoty ještě nekončí nastavení tohoto registru. Ne vždy lze totiž nastavit jakoukoliv přenosovou rychlost, protože výše vypočtená hodnota nemusí být celé číslo. Tím vzniká odchylka od správné hodnoty rychlosti. Podle výrobce je pro výše uvedenou přenosovou rychlost a frekvenci krystalu chyba 0,2 %, což je velmi nízká hodnota, a proto je umožněno použít danou přenosovou rychlost (38400 Bd).

Mikrokontrolér je schopen generovat 3 druhy přerušení: příjem dokončen, vysílání dokončeno, prázdný vysílací vyrovnávací registr. Příznaky přerušení jsou obsaženy v registru UCSR0A. Přerušení se povoluje v registru UCSR0B.

Příznak od prázdného vysílacího vyrovnávacího registru je nastaven do jedničky vždy, když UDR0 neobsahuje žádná data. Nulový je tehdy, pokud UDR0 obsahuje data, která ještě nebyla přesunuta do vysílacího posuvného registru. Tento příznak se maže až po zápisu nové hodnoty do UDR0. Jestliže data opustí vysílací posuvný registr a také datový registr UDR0 je prázdný, dojde k nastavení příznaku přerušení od dokončení vysílání. Příznak se nuluje v přerušovací rutině. Příznak přerušení od dokončeného příjmu je nastaven do jedničky pokud jsou v přijímacím bufferu nepřčtená data. Příznak se vynuluje až po přečtení UDR0. Pokud k tomuto přečtení nedojde, je přerušení generováno neustále.

Povolení vysílání a příjmu musí být nastaveno v registru UCSR0B. Samotné zahájení vysílání je poté zápisem do UDR0.

Nastavení řídicích registrů je následující:

UCSR0A – řídicí registr A

bit	7	6	5	4	3	2	1	0
	RXC0	TXC0	UDRE0	FE0	DOR0	UPE0	U2X0	MPCM0
read	0/1	0/1	0/1	0/1	0/1	0/1	0	0

- RXC0 – příznak dokončení příjmu
- TXC0 – příznak ukončení vysílání
- UDRE0 – příznak prázdného vysílacího bufferu (nevyužito)
- FE0 – příznak chyby rámce (špatný stop bit)
- DOR0 – příznak chyby přetečení přijímacího bufferu
- UPE0 – příznak chybné parity přijatých dat
- U2X0 – nastavení dvojnásobné rychlosti přenosu (nevyužito)
- MPCM0 – mód multiprocessorové komunikace (nevyužito)

UCSR0B – řídicí registr B

bit	7	6	5	4	3	2	1	0
	RXCIE0	TXCIE0	UDRIE0	RXEN0	TXEN0	UCSZ02	RXB80	TXB80
set	1	1	0	1	1	0	0	0

- RXCIE0 – povolení/zákaz přerušeni od dokončení příjmu
- TXCIE0 – povolení/zákaz přerušeni od dokončení vysílání
- UDRIE0 – povolení/zákaz přerušeni od prázdného vysílacího vyrovnávacího bufferu
- RXEN0 – povolení/zákaz příjmu dat
- TXEN0 – povolení/zákaz odesílání dat
- RXB80 – 9. datový bit pro příjem (nevyužito)
- TXB80 – 9. datový bit pro odesílání (nevyužito)

UCSR0C – řídicí registr C

bit	7	6	5	4	3	2	1	0
	-	UMSEL0	UPM01	UPM00	USBS0	UCSZ01	UCSZ00	UCPOL0
set		0	0	0	0	1	1	0

- UMSEL0 – přepínání mezi asynchronní a synchronní komunikací (nevyužito)
- UPM01 a UPM00 – nastavení parity (nevyužito)
- USBS0 – počet stop bitů (0 – 1 bit, 1 2 bity)
- UCSZ00 až UCSZ02 – nastavení formátu dat (8 bitů)
- UCPOL0 – nastavení hodin pro synchronní komunikaci (nevyužito)

3.3 Další periférie

Další využité periférie jsou A/D převodník, vnitřní paměť EEPROM a samozřejmě vstupně/výstupní porty.

Vstupně/výstupní porty jsou 8-bitové. Každý pin portu je chráněn diodami proti zemi a proti napájecímu napětí. Dalším vybavením je interní pull-up rezistor, který může a nemusí být aktivní. Každý port má tři ovládací registry (x označuje port, např. PORTA): směrový registr DDRx (Data Direction Register), datový registr PORTx (Port Data Register) a registr PINx (Port Input Pins Address) nesoucí informaci o skutečné log. úrovni pinů portu. Konkrétním nastavením vstupně/výstupních portů se nebudu zabývat, protože se jedná o běžné operace.

Paměť EEPROM slouží k ukládání dat pro parametrizování řídicí jednotky. Její předností je, že data uchovává i po přerušení napájení. Ve výsledku to znamená, že se jednotka nebude muset při každém zapnutí znovu parametrizovat. Obsluha paměti je velmi jednoduchá. Registry EEARH a EEARL (EEPROM Address Register) slouží k nastavení adresy paměti, EEDR (EEPROM Data Register) je vlastní datový registr, EECR (EEPROM Control Register) řídí operace s paměti – popis tohoto registru je níže.

EECR – řídicí registr

bit	7	6	5	4	3	2	1	0
					EERIE	EEMWE	EEWE	EERE
set					0/1	0/1	0/1	0/1

- EERIE – povolení/zákaz přerušení po dokončení čtení z paměti
- EEMWE – povolení zápisu do paměti
- EEWE – zahájení zápisu do paměti
- EERE – zahájení čtení z paměti

A/D převodník je používán k digitalizaci vstupních veličin. Mikrokontrolér Atmega64 je vybaven 10-bitovým A/D převodníkem s 8 kanály. Čtyři kanály jsou však určeny pro sériové programování přes rozhraní JTAG. Ze zbývajících čtyř kanálů jsou použity tři (převod proudu, napětí a žádané hodnoty). Výsledek převodu je pak nutné vynásobit určitou konstantou, abychom dostali vhodnou podobu převáděné veličiny. K ovládání A/D převodníku slouží registry: ADMUX (ADC Multiplexer Selection Register), ADCSRA a ADCSRB (ADC Control and Status Register). Výsledek převodu je uložen v 10-bitovém registru ADCW (opět rozdělen na dva 8-bitové, ADCH a ADCL). Generování přerušení od ukončení převodu není využíváno. Nastavení registrů je níže.

ADMUX – řídicí registr

bit	7	6	5	4	3	2	1	0
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0
set	0	0	0	0/1	0/1	0/1	0/1	0/1

- REFS1 a REFS0 – výběr napěťové reference
- ADLAR – zarovnání výsledku doprava/doleva v ADCW
- MUX0 až MUX4 – výběr kanálu

ADCSRA – řídicí registr A

bit	7	6	5	4	3	2	1	0
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0
set	1	0	0	0	0	1	1	0

- ADEN – povolení/zákaz A/D převodníku
- ADSC – zahájení převodu zápisem1, po dokončení převodu se vynuluje
- ADATE – zapíná automatický převod na událost (nevyužito)
- ADIF – příznak ukončení převodu
- ADIE – povolení/zákaz přerušení po skončení převodu
- ADSP0 až ADSP2 – nastavení děličky taktovacího kmitočtu pro ADC

ADCSR_B – řídicí registr B

bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	ADTS2	ADTS1	ADTS0
set						0	0	0

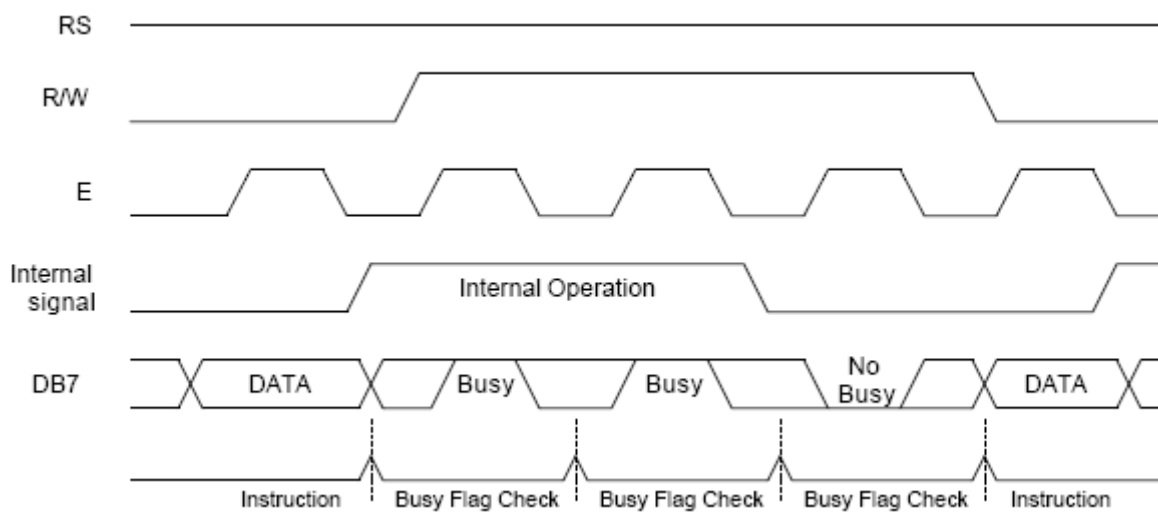
- ADTS0 až ADTS2 – volba spouštěcího signálu (nevyužito)

4 OBSLUHA PERIFERNÍCH ZAŘÍZENÍ

V této kapitole budou popsány obsluhy periferních zařízení, které přímo nesouvisí s regulační strukturou (ta bude popsána v další kapitole). Jde o zobrazování dat na displeji, komunikaci a zpracování dat, které budou sloužit jako vstupy do regulační struktury nebo jako výstupy místního ovládání.

4.1 Zobrazovací jednotka

Komunikace se zobrazovací jednotkou (displejem) probíhá podle Obr. 4.1. Ovládací vstup RS určuje, zda bude displej přijímat data (log. 1) nebo vykonávat instrukce (log. 0). Na vstupu R/W se nachází stále log. 0 (trvale uzemněn), protože možnost čtení informací z displeje není využita. Podle obrázku musí být generován taktovací signál E, aby byla data ze sběrnice do displeje načtena. Displej reaguje na náběžnou hranu tohoto signálu. Generování signálu E je zajištěno při každém odeslání dat nebo příkazu displeji formou změny hodnoty na výstupním pinu.



Obr. 4.1 – Komunikace s displejem

Displej obsahuje dvě paměti CGRAM (Character Generator Random Acces Memory) a DDRAM (Display Data Random Acces Memory). Do paměti DDRAM se ukládá pozice znaku na displeji. Adresy políček na displeji PC1602-F (2x16 znaků) jsou standardní, tj. 1. řádek 0x00 až 0x0F, 2. řádek 0x40 až 0x4F. Adresa je 7-bitová a nachází se na ní kód znaku. Je používána znaková sada s kódováním ASCII (American Standard Code for Information Interchange). Znak má velikost 5x8 bodů. Pokud nestačí znaková sada, je možné definovat až 8 vlastních znaků, které jsou uloženy v paměti CGRAM (není využito). Na Obr. 4.2 je znaková sada displeje. Vyšší bity (DB7 až DB4) určují sloupec a zbylé čtyři bity (DB3 až DB0) určují pozici na řádku.

Upper 4 bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
xxxx0000				0	@	P	`	P				-	9	3	α	ρ	
xxxx0001	(2)		!	1	A	Q	a	q				。	7	チ	4	ä	q
xxxx0010	(3)		"	2	B	R	b	r				「	イ	ツ	×	β	θ
xxxx0011	(4)		#	3	C	S	c	s				」	ウ	テ	ε	∞	
xxxx0100	(5)		\$	4	D	T	d	t				、	エ	ト	フ	μ	Ω
xxxx0101	(6)		%	5	E	U	e	u				・	オ	ナ	1	σ	ü
xxxx0110	(7)		&	6	F	V	f	v				ヲ	カ	ニ	ヨ	ρ	Σ
xxxx0111	(8)		'	7	G	W	g	w				ア	キ	ヌ	ラ	q	π
xxxx1000	(1)		<	8	H	X	h	x				イ	ク	ネ	リ	フ	×
xxxx1001	(2)		>	9	I	Y	i	y				ウ	ケ	ル	ル	イ	y
xxxx1010	(3)		*	:	J	Z	j	z				エ	コ	ン	レ	j	≠
xxxx1011	(4)		+	;	K	C	k	c				オ	サ	ヒ	ロ	*	π
xxxx1100	(5)		,	<	L	¥	l	l				ハ	シ	フ	ワ	φ	π
xxxx1101	(6)		-	=	M	J	m	j				ユ	ズ	ヘ	ン	ト	÷
xxxx1110	(7)		.	>	N	^	n	→				ヨ	セ	ホ	ド	ñ	
xxxx1111	(8)		/	?	O	_	o	€				ウ	ツ	マ	°	ö	■

Obr. 4.2 – Znaková sada

Princip ovládání displeje spočívá v zasílání instrukcí a dat řadiči. Instrukční sada řadiče pro displej je uvedena v tabulkách Tab. 4.1 a Tab. 4.2. Lze mazat displej, nastavovat polohu kurzoru, atd. Po odeslání příkazu je vždy nutné vyčkat určitou dobu, kterou řadič LCD potřebuje na zpracování. Zdrojový kód pro ovládání displeje byl použit z bakalářské práce autora [8].

Instrukce	RS	R/W	Data (DBx)								Doba provedení	
			7	6	5	4	3	2	1	0		
Vymaž displej	0	0	0	0	0	0	0	0	0	0	1	1,53 ms
Kurzor na začátek	0	0								1	x	1,53 ms
Směr posunutí	0	0						1	I/D	SH		39 μs
Zap./vyp. displeje	0	0					1	D	C	B		39 μs
Posuny	0	0				1	S/C	R/L	x	x		39 μs
Inicializace	0	0			1	DL	N	F	x	x		39 μs
Zápis do CGRAM	0	0		1	adresa v CGRAM							39 μs
Zápis do DDRAM	0	0	1	adresa v DRAM							39 μs	
Čtení příznaku	0	1	BF	adresa v DRAM							0 μs	
Zápis dat do paměti	1	0	data								39 μs	
Čtení dat z paměti	1	1	data								39 μs	

Tab. 4.1 – Instrukční sada

I/D	směr posuvu kurzoru	0-	vlevo	1-	vpravo
SH	posuv textu	0-	zakázán	1-	povolen
D	displej	0-	vypnut	1-	zapnut
C	kurzor	0-	vypnut	1-	zapnut
B	blikání kurzoru	0-	vypnuto	1-	zapnuto
S/C	posun	0-	kurzoru	1-	textu
R/L	posun	0-	vlevo	1-	vpravo
DL	komunikace	0-	4-bitová	1-	8-bitová
N	počet řádků	0-	jeden	1-	dva
F	velikost fontu	0-	5x8 bodů	1-	5x10 bodů
BF	příznak	0-	příjem povolen	1-	řadič zaneprázdněn

Tab. 4.2 – Význam jednotlivých bitů z Tab. 4.1

4.2 Komunikační protokol

Specifikace komunikace mezi řídicí jednotkou a počítačem je dána komunikačním protokolem. Komunikační protokol je založen na determinovaném přístupu ke sběrnici. Počítač je v komunikační hierarchii nadřazen řídicí jednotce, a proto on určuje, co se bude na sériové lince dít. Formát vysílaných dat je UART (Universal Asynchronous serial Receiver and Transmitter), 8 datových bitů. 1 stop bit, žádná parita.

Struktura vysílaných zpráv se liší podle směru komunikace. Při odesílání zpráv z počítače do řídicí jednotky je vždy na počátku zprávy odeslán jeden bajt mající význam identifikátoru zprávy. Tento identifikátor určuje, jaká operace bude následovat, zda-li se v mikrokontroléru vykoná nějaká operace nebo budou přijímána další data. Identifikátor tak má v některých případech funkci příkazu. Většina přijatých zpráv je zpětně potvrzována.

Pro odesílání většího množství dat jsou proměnné ukládány do struktury a ta poté jako celek odeslána. Na obou stranách komunikace pak musí být dodržen formát struktury. To znamená, že délka dat musí být stejná a známá, aby byla načtena všechna data a ne jen jejich část. Část zdrojového kódu byla použita z výuky předmětu Aplikace mikroprocesorů [9].

Komunikační protokol je uveden v Tab. 4.3 (Pozn. 2Q = dvoukvadrantový měnič). Potvrzování přijatých dat je ve formě odeslání znaku 'O'. V případě zaslání identifikátoru 'k', může být místo znaku 'O' odeslán kód chyby, pokud k ní samozřejmě dojde. Při požadavku na odeslání parametrů je odesílána celá struktura ve stejném formátu v jakém byla přijata. U dvou stavů není zpráva potvrzována znakem 'O', protože by došlo ke kolizi v identifikaci přijímaných dat. Pokud jsou data periodicky odesílána z řídicí jednotky do počítače je jejich velikost 4x2B, kdy každé dva bajty reprezentují odesílanou hodnotu.

Data odesílaná z počítače				Odpověď řídicí jednotky	
identif. (1B)	význam identifikátoru	další data	vel.	reakce	vel.
'k'	komunikace OK	ne	-	potvrzení / chyba	1B
'p'	příjem parametrů	ano	37B	potvrzení	1B
'h'	hodnota z počítače	ano	2B	bez odpovědi	-
's'	STOP regulace	ne	-	potvrzení	1B
'z'	START regulace	ne	-	potvrzení	1B
'x'	zahájení per. odesílání dat	ne	-	bez odpovědi	-
'y'	ukončení odesílání dat	ne	-	potvrzení	1B
'r'	reverzace vpravo (pouze 2Q)	ne	-	potvrzení	1B
'l'	reverzace vlevo (pouze 2Q)	ne	-	potvrzení	1B
'n'	odeslání parametrů	ne	-	data	37B

Tab. 4.3 – Komunikační protokol

4.3 Vstupní obvody

Všechny hodnoty z A/D převodníku jsou filtrovány, protože ve vzorcích signálu je většinou obsažen šum, který je potřeba omezit. Vytvořený filtr používá algoritmus tzv. „klouzavého průměru s exponenciálním zapomínáním“. Tento algoritmus nevyžaduje uchovávání všech vzorků v paměti. Nejstarší vzorek je nahrazen jeho odhadem. Nejlepší odhad je poslední hodnota průměru. Pro výpočet průměru pak postačí pouze poslední vzorek. Algoritmus je popsán rovnicí (5).

$$avg = avg + \alpha \cdot (vzorek - avg), \text{ kde} \quad (5)$$

avg – průměrná hodnota (výsledek filtrace)

α – váha filtrace

vzorek – hodnota z A/D převodníku

Dalším softwarovým zpracováním vstupních dat je zjišťování polarity proudu (resp. napětí). Jak bylo dříve zmíněno A/D převodník pracuje pouze s unipolárními hodnotami musí být polarita zjišťována až samotným softwarem. Celý 10-bitový rozsah je rozdělen na dvě poloviny, z nichž každá představuje jednu polaritu. Nulová hodnota proudu (napětí) tak má hodnotu 512 (z A/D převodu).

5 REGULAČNÍ STRUKTURA

5.1 Regulační obvod

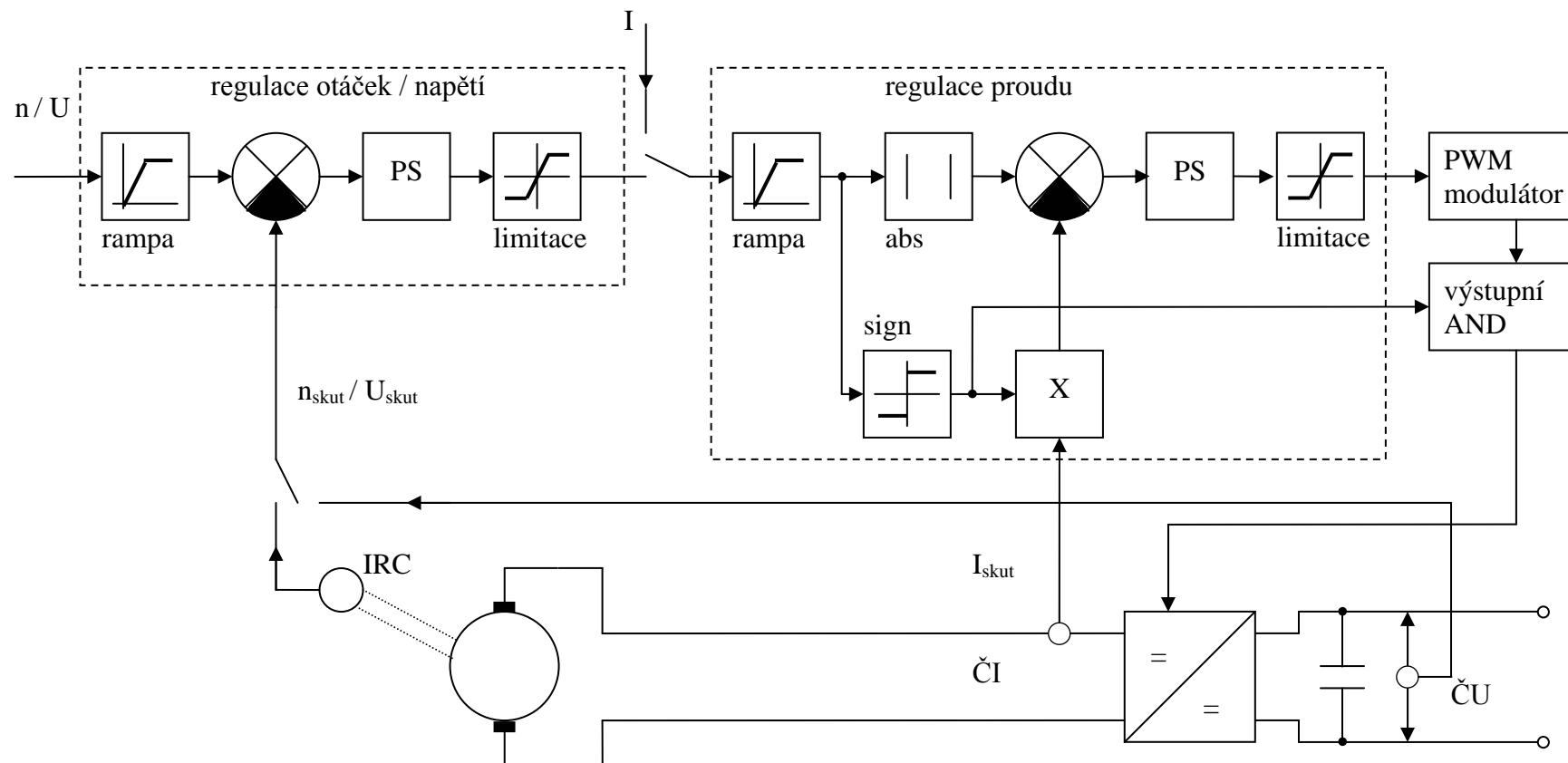
Základem celé regulační struktury je její regulační obvod (Obr. 5.1). Skládá se z regulované soustavy, která je tvořena čtyřkvadrantovým měničem a stejnosměrným motorem s permanentními magnety. Této soustavě je nadřazen PWM modulátor s výstupní jednotkou, tvořenou hradly AND. Modulátor zajišťuje tvorbu PWM signálu podle hodnoty, získané z nadřazené části. Touto nadřazenou částí je proudová smyčka. Tato část regulačního obvodu je jeho nejdůležitější částí, protože představuje základ celé řídicí jednotky. Proudové smyčce může a nemusí být nadřazena ještě další smyčka (otáčková nebo napěťová). Zpětná vazba je zajišťována vždy proudovým čidlem LEM a v případě použití další smyčky ještě inkrementálním čidlem nebo napěťovým čidlem.

Celá regulační struktura lze parametrizovat. To znamená, že jednotlivé parametry regulačního obvodu mohou být libovolně měněny. Regulovat tak lze několik veličin. Jedná se o regulaci proudu, otáček a napětí stejnosměrného meziobvodu. Regulace proudu se však dá považovat také za regulaci momentu, neboť moment je téměř lineárně závislý na proudu (pro stejnosměrné motory s permanentními magnety).

Detailnější popis pokračuje dle obrázku zleva doprava.

Nadřazenou smyčku tvoří regulace otáček nebo napětí. Pro tyto dvě veličiny je vytvořena pouze jedna regulační smyčka. V důsledku to tedy znamená, že lze regulovat pouze jednu z těchto dvou veličin (volba v parametrizování). Jak regulace otáček tak regulace napětí využívá jeden PS regulátor. Rozdíl nastává pouze při parametrizování, kdy pro každou regulovanou veličinu jsou nastaveny jiné parametry regulátoru. Podle zvolené regulované veličiny je samozřejmě na rozdílový člen smyčky přivedena správná skutečná hodnota, tzn. regulace otáček – signál z IRC čidla, regulace napětí – signál z napěťového čidla LEM. Regulační smyčka je tedy tvořena rampou (brání příliš strmému nárůstu požadované hodnoty), rozdílovým členem, PS regulátorem a limitačním členem (omezuje hodnoty pro další smyčku (proudovou)).

Je-li v parametrizování volba regulace otáček nebo napětí, jsou výsledky z této regulační smyčky předávány následující smyčce. Pokud je však regulován pouze proud, představuje tato část regulačního obvodu vstupní místo pro žádanou hodnotu proudu.



Obr. 5.1 – Regulační obvod

Další a nejdůležitější částí regulačního obvodu je proudová smyčka. Tato smyčka je taktéž tvořena rampou rozdílovým členem, PS regulátorem a limitačním členem, ale na rozdíl od předcházející smyčky jsou zde navíc další specifika. Na počátku proudové smyčky se nachází tzv. rampa. Dalšími členy jsou už jen čistě matematické funkce: absolutní hodnota, signum a násobení. Účelem těchto tří členů je vhodně upravit jak žádanou tak skutečnou hodnotu proudu. Navíc se zde získává hodnota (spíše příznak) pro správnou činnost výstupních hradel AND. Důvodem, proč byla tato část proudové smyčky takto upravena, je použití unipolárního řízení. Účelem těchto přepočtů je to, že pokaždé bude mít regulační odchylka stejnou polaritu, ať budeme regulovat proud v kladném nebo záporném smyslu. Bližší pohled na princip výpočtu je vidět v Tab. 5.1 (hodnoty jsou pouze ukázkové). Výstupem z proudové smyčky je již přímo hodnota pro PWM modulátor.

žádaná hodnota	skutečná hodnota	regulační odchylka
10	12	-2
10	8	2
10	-12	22
10	-8	18
-10	-12	-2
-10	-8	2
-10	12	22
-10	8	18

Tab. 5.1 – Princip výpočtu regulační odchylky u proudové smyčky

Všechny výše uvedené součásti regulačního obvodu jsou implementovány do řídicího softwaru. Požadavkem na řídicí software je to, aby každá smyčka byla opakována se stejnou periodou. Regulační obvod je jednou z nejdůležitějších součástí celého softwaru.

5.2 PS Regulátor

PS regulátor (Proporcionálně Sumační regulátor) je základním členem regulační smyčky. Jedná se o diskretní podobu spojitého PI regulátoru (Proporcionálně Integrovního). V časové oblasti je regulátor (PI) popsán rovnicí (6). Diskretní regulátor však nevyhodnocuje data spojitě, nýbrž v diskretních okamžicích $t = k T_S$ ($k = 0, 1, 2, \dots$), kde T_S je perioda vzorkování. Perioda vzorkování by měla být dostatečně malá, protože při delší době vzorkování by se ztrácely informace o velikosti regulované veličiny mezi okamžiky vzorkování. Diskretizace spojitého PI regulátoru se provede náhradou integrace sumací (zpětná obdélníková metoda). Zápis regulátoru v diskretní podobě je poté popsán rovnicí (7).

$$u(t) = K_p \cdot e(t) + \frac{K_p}{T_I} \int_0^t e(\tau) d\tau \quad (6)$$

$$u(k) = K_p \cdot e(k) + K_I \cdot \sum_{i=1}^k e(i), \text{ kde} \quad (7)$$

$$K_I = \frac{K_p \cdot T_S}{T_I} - \text{integrační zesílení}$$

K_p – proporcionalní zesílení

T_S – perioda vzorkování

T_I – integrační časová konstanta

e – regulační odchylka

k – číslo vzorku ($k = 0, 1, 2, \dots$)

Zápis rovnice (7) se nazývá polohový algoritmus PS regulátoru. Tento algoritmus vypočítává v každém diskretním časovém okamžiku $k T_S$ aktuální hodnotu akční veličiny $u(k)$. Tento algoritmus nepotřebuje ukládat jednotlivé hodnoty regulačních odchylek, protože ty jsou již zpracovány ve formě sumy. To znamená, že k výpočtu je potřeba pouze aktuální hodnota regulační odchylky a suma všech odchylek. Proporcionalní i sumační složka algoritmu mají stejné vlastnosti jako obdobné složky spojitého regulátoru. Na počátku převládá vliv proporcionalní složky a s narůstajícím počtem vzorků převládá vliv sumační složky. V důsledku tedy PS regulátor odstraňuje trvalou regulační odchylku a zlepšuje stabilitu.

Při realizaci PS regulátoru je nutné dbát na různá omezení. Jednotlivé složky regulátoru (P, S) by se měly omezovat s ohledem na možné přetečení konečného výsledku, tzn. limitovat již mezivýsledky. Další operací je limitace akčního zásahu. Výstupem z regulátoru je komparační hodnota pro PWM modulátor, která musí ležet v daném intervalu. Výsledek algoritmu může však být mnohem větší (menší) číslo mimo tento interval, proto je nutno provádět limitaci.

Velký význam má omezení integrační složky (antiwind-up). Při dosažení limitní hodnoty akční veličiny $u(k)$ i nadále narůstá sumační složka při nezměněné polaritě regulační odchylky $e(k)$, aniž by se to projevilo na výstupní hodnotě $u(k)$, neboť ta je limitována. Tato situace nastane běžně, např. při požadavku na velkou hodnotu regulované veličiny $y(k)$. To se projeví velkým překmitem nad/pod žádanou hodnotu na přechodové charakteristice. Řešením je pozastavit sumaci, jestliže je dosažena limitní hodnota akční veličiny $u(k)$, dokud se nezmění polarita regulační odchylky $e(k)$, která poté dovolí zmenšení sumační složky. To lze u polohového algoritmu zajistit pomocí podmínky. Sumaci se musí zabránit v těchto dvou případech (u_{MAX} , u_{MIN} jsou limitní hodnoty akční veličiny):

- $u(k) > u_{MAX}$ a $e(k) > 0$
- $u(k) < u_{MIN}$ a $e(k) < 0$

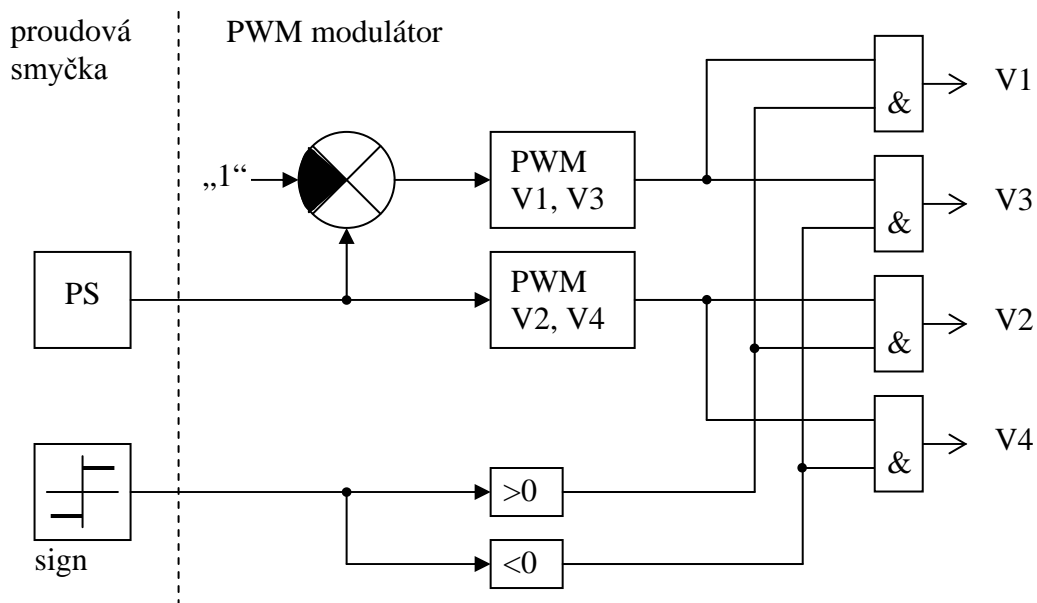
Výsledná podoba softwarové realizace je dána skloubením polohového algoritmu popsaného rovnicí (6) a výše uvedených nastavení. Pro správnou práci regulátoru je pak nutné jej správně parametrizovat, tj. vhodně nastavit konstanty regulátoru. [9]

5.3 PWM modulátor

Posledním článkem v regulačním řetězci před vstupem do výkonové části (měniče) je výstupní část řídicí jednotky tvořená PWM modulátorem a výstupní logikou. PWM modulátor je realizován v rámci mikrokontroléru využitím jeho schopností generování PWM signálu. Výstupní logiku tvoří čtyři hradla typu AND.

Pro řízení měniče je třeba mít k dispozici čtyři výstupní piny, s různou hodnotou signálu PWM. Mikrokontrolér však disponuje maximálně třemi takovými piny (komparační jednotka Timer/Counter1), proto je potřeba udělat jistá opatření. V hardwarové realizaci to znamená, že použijeme dvě komparační jednotky, na jejichž výstupních pinech se nacházejí dva různé PWM signály. Dále z mikrokontroléru použijeme ještě dva klasické výstupní piny. Všechny čtyři piny jsou pak přivedeny na čtyři hradla AND, která zajišťují přenos PWM signálu na všechny spínací prvky měniče.

Každý PWM signál je přiveden na dvě hradla a taktéž klasické výstupní piny jsou přivedeny na dvě hradla, avšak s rozdílným signálem PWM. Podle spínacího algoritmu je zajištěn přenos PWM signálu na čtyři spínací prvky měniče., což bylo záměrem. Blokové schéma PWM modulátoru včetně výstupní logiky je na Obr. 5.2.



Obr. 5.2 – PWM modulátor a výstupní logika

Princip spínacího algoritmu je odvozen od požadovaného režimu řízení (motorický, generátorický). Zvolme, že pulzní řízení se bude v motorickém režimu uskutečňovat pomocí tranzistorů V1 a V3 (značení dle Obr. 1.1). Tranzistory V2 a V4 budou v generátorickém režimu použity pro pulzní řízení, avšak v motorickém režimu budou naplno otevřeny (hodnota PWM 100%). Podle polaroty požadovaného proudu jsou spínány příslušné dvojice tranzistorů V1, V2 nebo V3, V4. Polarita je získána z funkce signum, která se nachází v proudové smyčce.

PWM komparátory mikrokontroléru požadují hodnoty v rozsahu 0 až 1023 (10 bitů). To by měl být tedy i rozsah výstupní hodnoty PS regulátoru, avšak jeho výstupní hodnota je v rozsahu 0 až 2047. První polovina rozsahu, tj. 0 až 1023, je využita pro první PWM komparátor určený pro tranzistory (V2, V4) pracující v generátorickém režimu. Po překročení hodnoty 1023 jsou již tyto tranzistory naplno otevřeny, hodnota je zalimitována. Druhá polovina rozsahu, tj. 1023 až 2047, slouží pro druhý PWM komparátor určený pro tranzistory (V1, V3) pracující v motorickém režimu. Samozřejmostí je přepočítání hodnoty na správný tvar požadovaný komparační jednotkou. Musí se odečíst hodnota plného otevření „1“, tj. 1023.

Tímto způsobem se dosáhne plynulého přechodu z generátorického do motorického režimu. Při přechodu z motorického do generátorického režimu je však nutné zajistit časovou rezervu, aby nedošlo k sepnutí obou tranzistorů v jedné větvi a tedy ke zkratu.

6 VÝSLEDNÝ ŘÍDÍCÍ SOFTWARE

Celý software byl naprogramován v jazyce C. K napsání kompletního softwaru bylo použito vývojové prostředí programu AVR Studio 4 od firmy Atmel. Při psaní softwaru byl ještě využit program CodeVision. Ten sloužil k rychlému nastavování registrů mikrokontroléru pomocí funkce CodeWizard. Vygenerovaný kód nastavení registrů byl poté aplikován ve vytvářeném softwaru. Pro překlad softwaru z jazyka C do assembleru byl použit kompilátor GCC.

Konečný software je rozdělen do několika souborů, které obsahují data vždy z dané oblasti, např. soubory určené pro ovládání komunikace, pro řízení regulační struktury, atd. Všechny tyto soubory jsou pak používány v jednom hlavním souboru (nachází se v něm funkce `main()`), kde se také odehrává hlavní struktura celého softwaru. Rozdělením programu do více souborů také umožnilo využít výhod odděleného překladu. To je výhodné zejména při ladění programu, kdy je překládána jen ta část, která byla měněna.

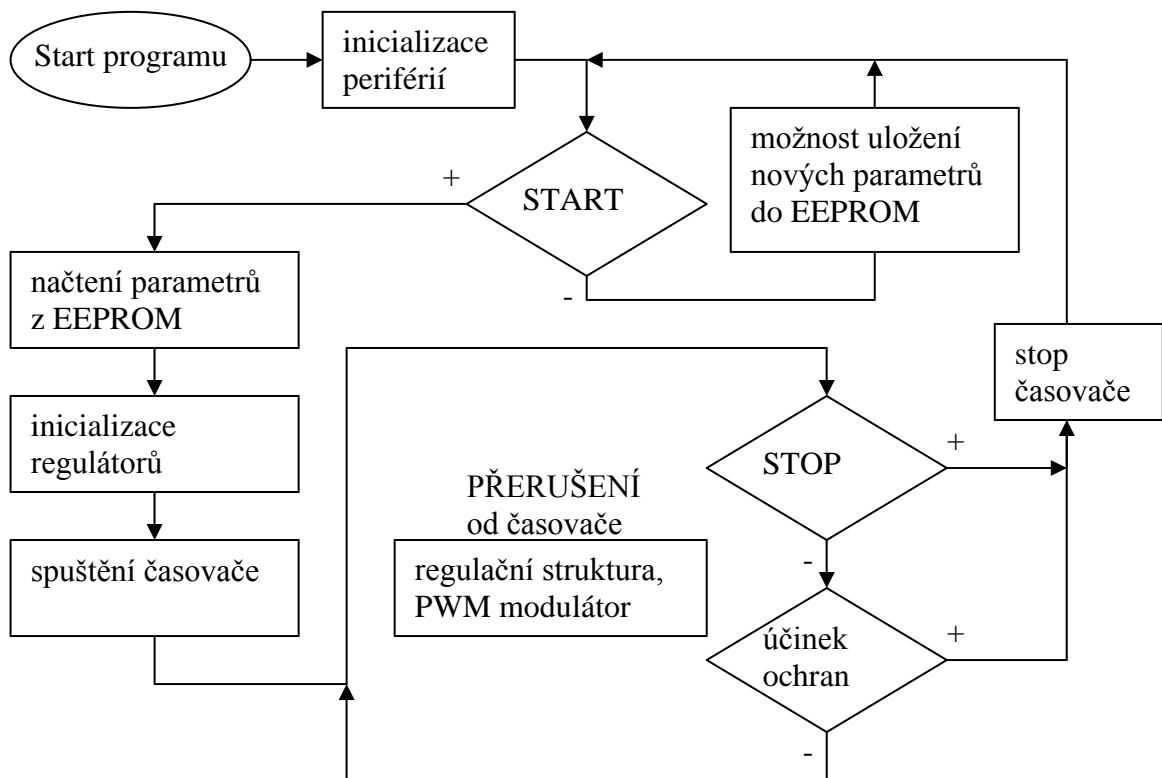
Software pro komunikaci s řídicí jednotkou (nebyl požadován v zadání) byl vytvořen ve vývojovém prostředí Delphi vedoucím práce Ing. Jiřím Šimánkem.

Veškeré výpočetní operace byly prováděny v celočíselném tvaru, aby se urychlila činnost. Všechny reálné hodnoty byly pro účely výpočtu převedeny (vyškálovány) na celočíselný tvar. Dále bylo využito při operacích násobení a dělení, kde to bylo možné, instrukce bitového posunu, která rovněž velmi zrychluje činnost programu. Rychlost výpočtu se nejvíce projeví v případě dělení, protože mikrokontroléry AVR nemají přímo instrukci pro dělení a výpočet se proto musí dělat složitějším a delším způsobem. Počet přerušení je co nejmenší (tři – přetečení časovače, příjem dokončen, odeslání dokončeno), aby nedošlo k narušování běhu regulační struktury.

6.1 Hlavní smyčka

Struktura hlavní části programu je na Obr. 6.1. Po zapnutí řídicí jednotky dojde k inicializaci periférií, aby bylo možné s jednotkou pracovat. Další část softwaru je poté uzavřena v nekonečné (hlavní) smyčce.

Činnost programu v hlavní smyčce lze rozdělit na čtyři etapy. První etapa je uzavřena ve smyčce a je určena pro prvotní ovládání řídicí jednotky. Ve druhé etapě dochází k samotnému nastavování celé regulační struktury. Třetí etapa, jež je opět tvořena smyčkou, slouží k vlastní regulaci. Ve čtvrté etapě dochází k vyhodnocení stavu po skončení předcházející etapy.



Obr. 6.1 – Vývojová struktura hlavní smyčky programu

6.2 Vedlejší smyčky

Po inicializaci periférií se program dostane do první smyčky. Ta je ukončována povelu START, který lze zadat místně nebo z počítače. Tato smyčka je především určena pro komunikaci s počítačem. V tuto chvíli je totiž nejvhodnější parametrizovat celou jednotku, tzn. nastavovat parametry regulátorů, volbu ovládání, ochrany atd. (Seznam všech parametrů je uveden v příloze.) Přijaté parametry jsou ukládány do paměti EEPROM. To znamená, že parametrizování není při každém zapnutí řídicí jednotky nutné.

V případě užívání řídicí jednotky pro dvoukvadrantový měnič lze nastavit směr otáčení, z počítače nebo místně přes dva volné piny mikrokontroléru (jejich umístění v řídicí jednotce je uvedeno v příloze).

Mezi oběma smyčkami se nachází inicializační část regulační struktury. Nejprve jsou z paměti EEPROM načteny všechny parametry, které jsou poté využívány během činnosti regulační struktury. V této části programu se nacházejí funkce inicializující PS regulátory a PWM modulátor. Jsou rovněž nulovány všechny potřebné proměnné určené k řízení, např. příznaky ochran. Na konci této inicializační části je zapnut časovač.

V režimu pro dvoukvadrantový měnič se v tomto místě programu nastavují piny pro stykače určené ke změně směru otáčení (reverzaci). Je rovněž kontrolována jejich správná činnost. Pokud dojde k špatnému sepnutí, je nastaven příznak chyby, který neumožňuje spuštění regulace.

Následuje další smyčka. Ta může být ukončena několika způsoby:

- povel STOP (místně nebo z počítače),
- účinkem ochrany,
- vznikem chyby.

V této smyčce se provádějí operace výpisu dat na displej (žádané a skutečné hodnoty regulované veličiny). Je zde prováděna softwarová ochrana regulované soustavy. V případě překročení povolené hodnoty proudu, otáček nebo napětí, dojde k zaúčinkování ochrany, která zajistí ukončení regulace (opuštění smyčky). Kontroluje se zda nedošlo k chybě budičů (informace přenášena propojovacím kabelem) nebo k poruše komunikace s počítačem, je-li využita. V případě vzniku chyby opět dojde k ukončení regulace.

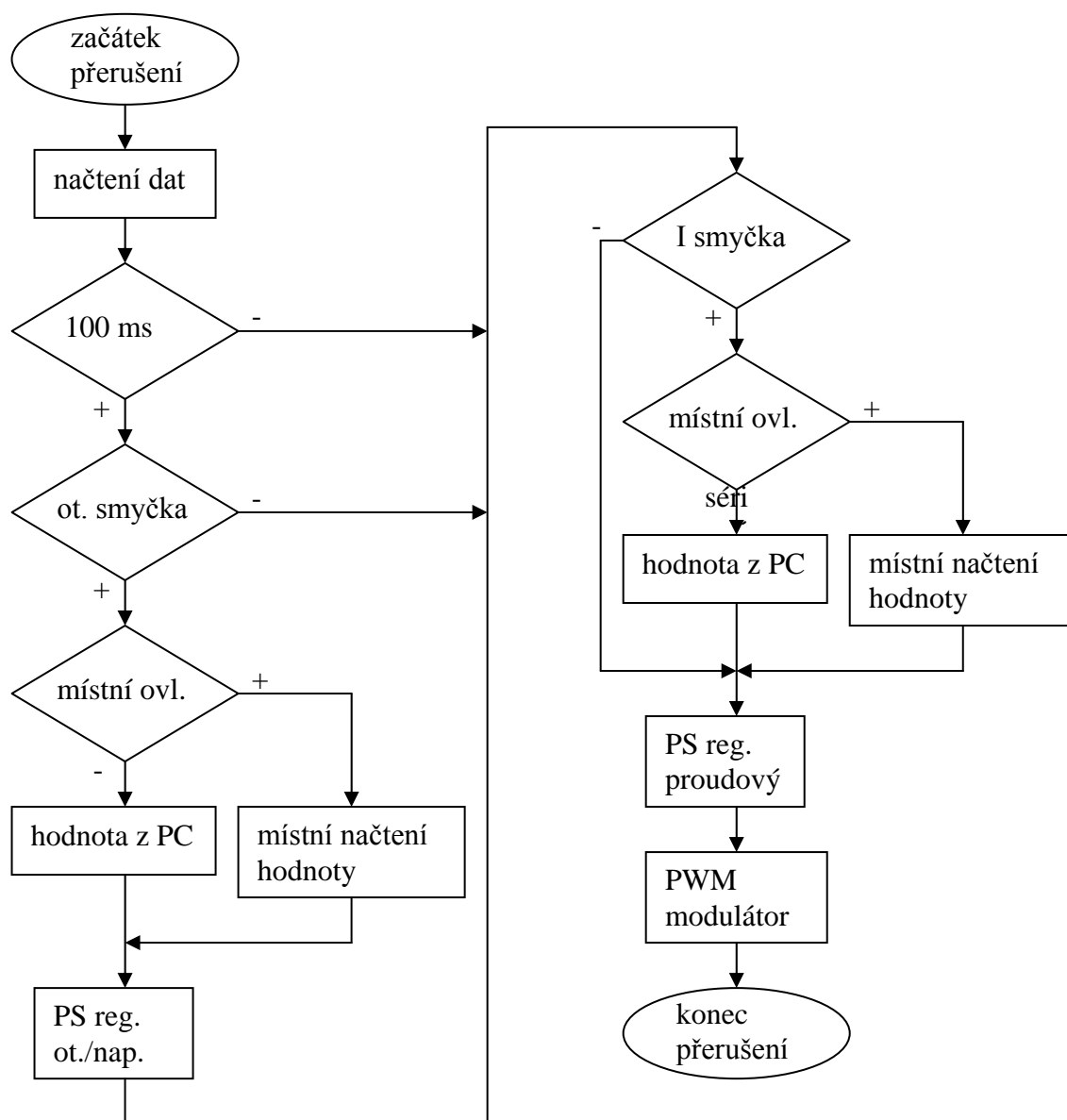
Během této smyčky je generováno přerušení od časovače, ve kterém probíhá celá regulační struktura. Zároveň je možné přijímat z počítače žádanou hodnotu regulované veličiny nebo odesílat vybraná data (dvě hodnoty) zpět do počítače.

V poslední etapě je zastaven časovač a zakázán výstup signálů z řídicí jednotky k budičům. Na displeji se zobrazí, jakým způsobem byla regulace ukončena, a zároveň se uloží kód tohoto způsobu ukončení pro případné odeslání do počítače, bude-li to vyžadováno. Ukončení této etapy je opět povel STOP (místně nebo z počítače) – vyčkávací smyčka.

6.3 Činnost programu v přerušení

Do rutiny přerušení je implementována celá regulační struktura z kapitoly 5.1. Přerušení je generováno každé 2 ms, což odpovídá vzorkovací periodě proudové smyčky. Pro otáčkovou/napěťovou smyčku je zvolena vzorkovací perioda 100 ms. Této hodnoty je dosaženo po každém 50-tém vyvolaném přerušení, není tedy použit další vhodný časovač, který v každém případě již není k dispozici. (Volný Timer/Counter2 je pouze 8-bitový a hodnoty 100 ms by se dosahovalo opět s počítadlem počtu přerušení.)

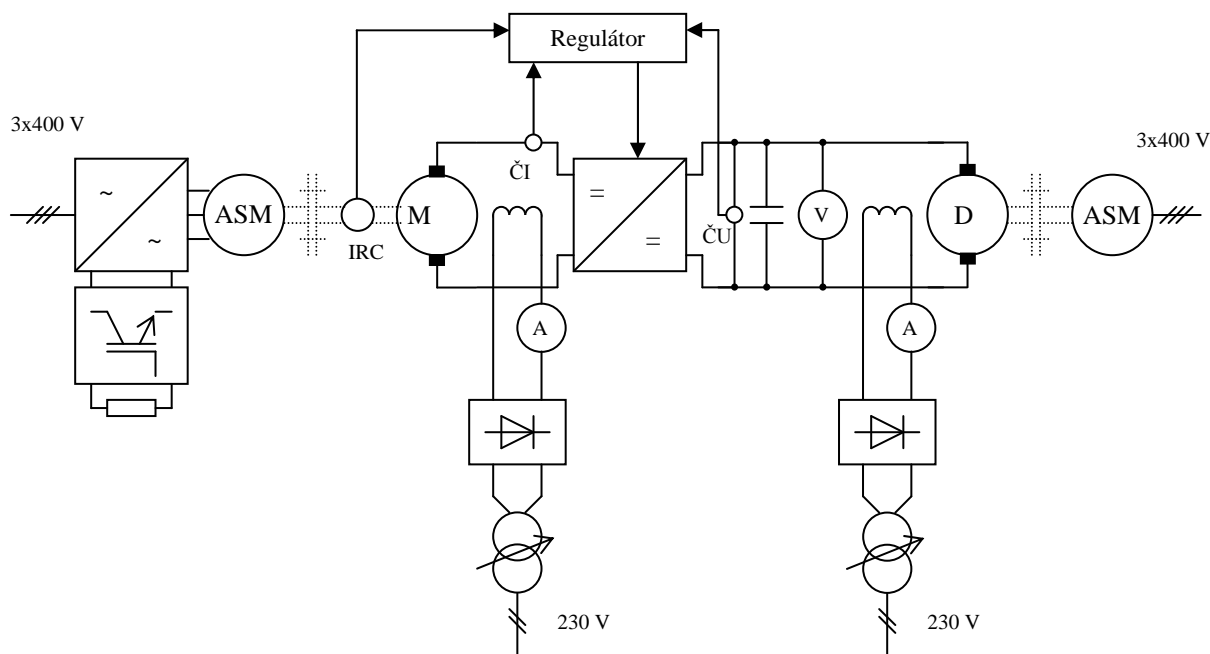
Na Obr. 6.2 je vidět vývojová struktura činnosti programu, která odpovídá regulačnímu obvodu. Při každém přerušení je vykonávána proudová smyčka, bez ohledu na to, zda se reguluje proud nebo jiná veličina. Nadřazená otáčková/napěťová smyčka je vykonávána pouze tehdy, je-li zvolena jako řídicí. Před každou smyčkou je načtena požadovaná hodnota regulované veličiny z místního ovládní nebo z počítače. V případě zadávání hodnot z počítače, je požadována periodičnost příchodu dat, i když žádaná hodnota může být stále stejná. Pokud není tato podmínka dodržena, dojde k vygenerování chyby komunikace. Poslední operací v přerušení je nastavení aktuálních hodnot pro PWM modulátor a výstupní logiku.



Obr. 6.2 – Vývojová struktura činnosti programu v přerušení

7 TESTOVÁNÍ ŘÍDÍCÍ JEDNOTKY

7.1 Testovací pracoviště



Obr. 7.1 – Schéma zapojení testovacího pracoviště

Na Obr. 7.1 je zapojení testovacího pracoviště pro řídicí jednotku (označena jako regulátor). V pravé části obrázku se nachází soustrojí pro tvorbu stejnosměrného napětí. Jelikož nebyl k dispozici stejnosměrný motor s permanentními magnety byl použit klasický cizebuzený motor, kterému bylo nutné zajistit buzení, protože buzení nemá být řídicí jednotkou ovládáno. Tento stejnosměrný motor je zatěžován asynchronním motorem řízeným frekvenčním měničem od firmy Siemens. Parametry stejnosměrného motoru a typy použitých čidel jsou níže:

- stejnosměrný cizebuzený motor:
 - jmenovitý výkon 1,5 kW
 - jmenovité otáčky 1450 min⁻¹
 - jmenovitý proud 6,5 A
 - jmenovité napětí 230 V
 - jmen. proud buzení 0,65 A
- proudové čidlo (LEM): LA 55-P
- napěťové čidlo (LEM): LV 25-P
- IRC čidlo (LARM a.s.): 303/1024 PB

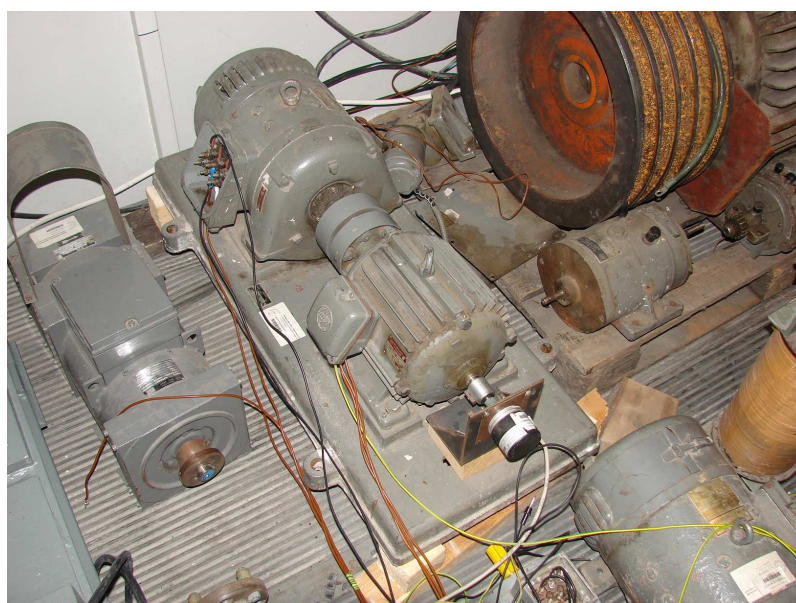
Testovací pracoviště je vidět na Obr. 7.2 a řízený stejnosměrný motor na Obr 7.3. V následujících částech kapitoly jsou zobrazeny průběhy z regulace proudu, otáček a napětí (Obr. 7.4 až Obr. 7.8). V grafech jsou zobrazovány žádané hodnoty, skutečné hodnoty (označeny hvězdičkou) a komparační hodnota pro PWM modulátor. Další průběhy viz příloha.) Parametry nastavení regulátorů jsou v Tab. 7.1.

	proudový	otáčkový	napěťový
Kp	1,95	0,25	0,5
Ti	5 ms	3000 ms	200 ms

Tab. 7.1 – Parametry regulátorů

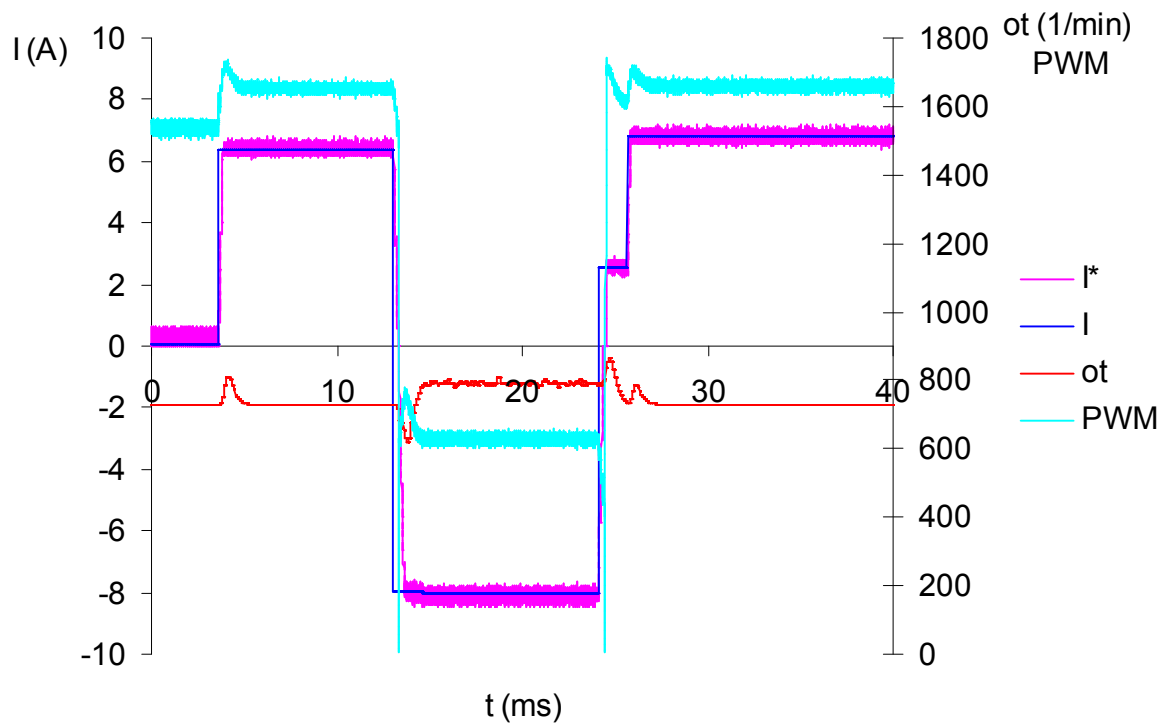


Obr. 7.2 – Testovací pracoviště

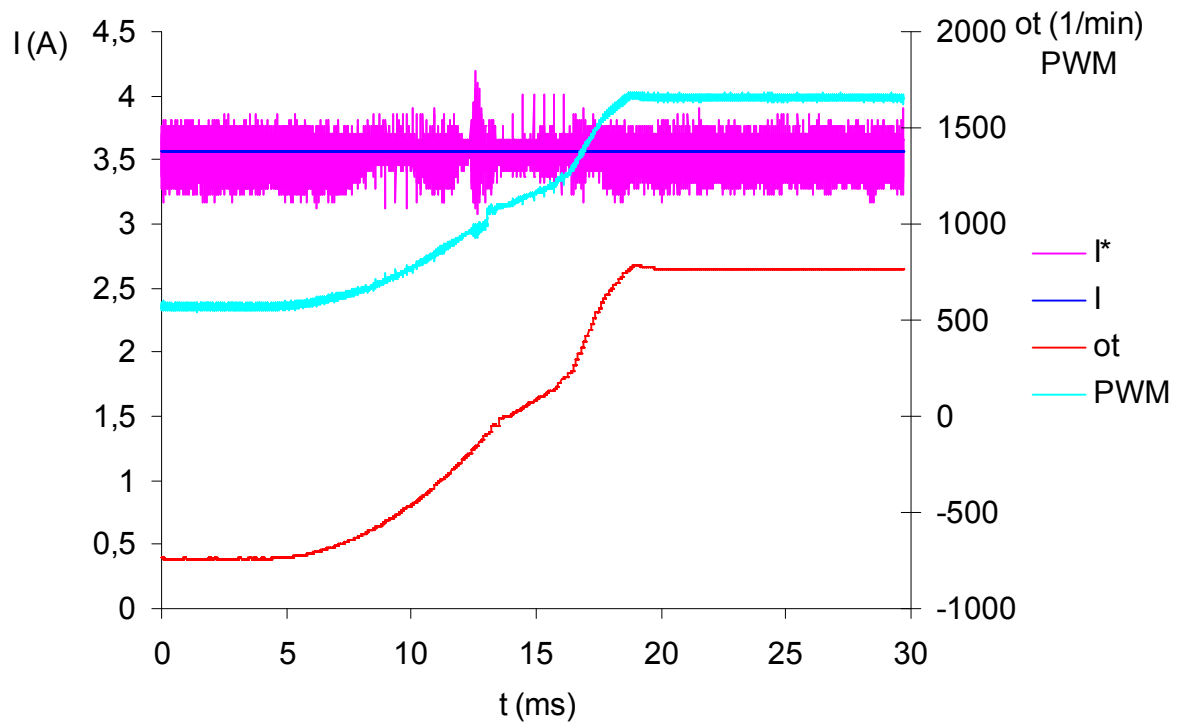


Obr. 7.3 – Řízený stejnosměrný motor (asynchronní motor jako zátěž)

7.2 Regulace proudu

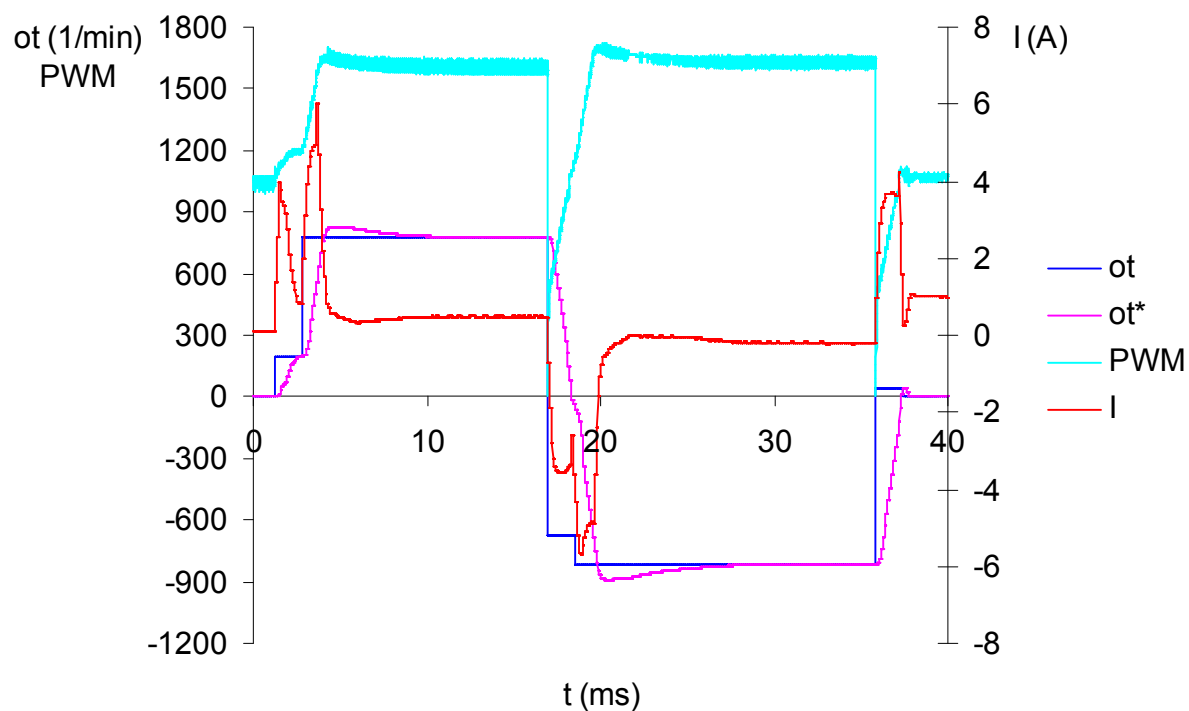


Obr. 7.4 – Regulace proudu (při konstantních otáčkách)

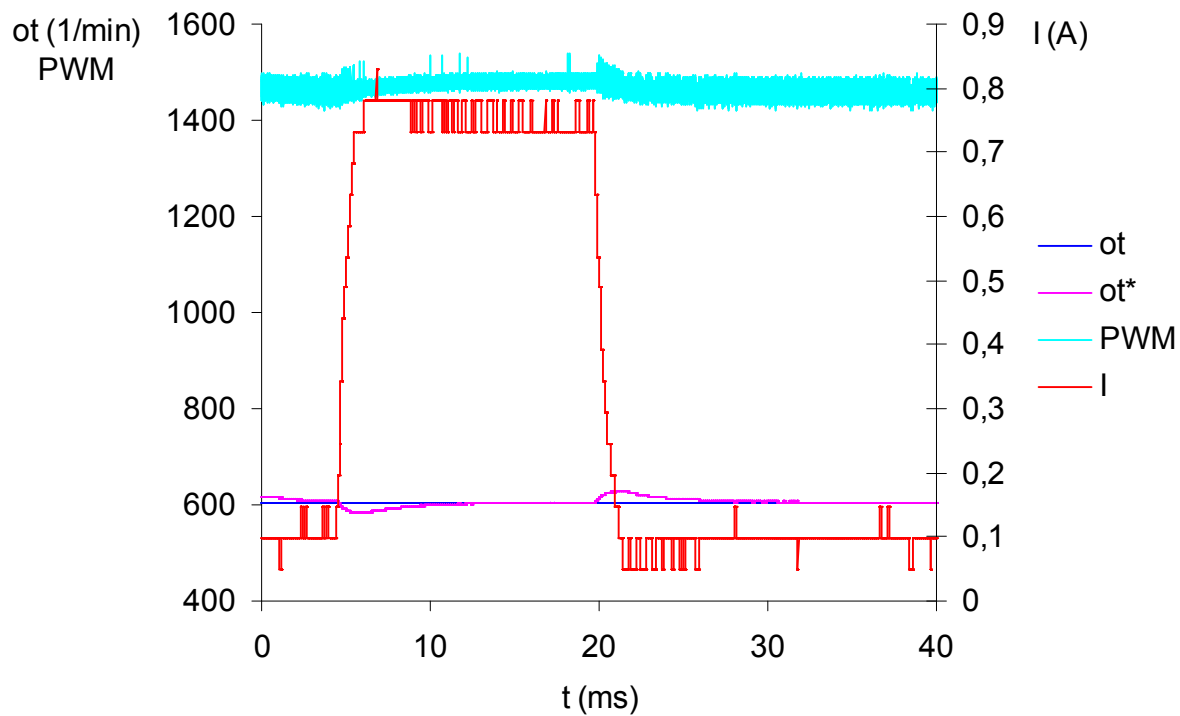


Obr. 7.5. – Regulace proudu se změnou zatížení

7.3 Regulace otáček

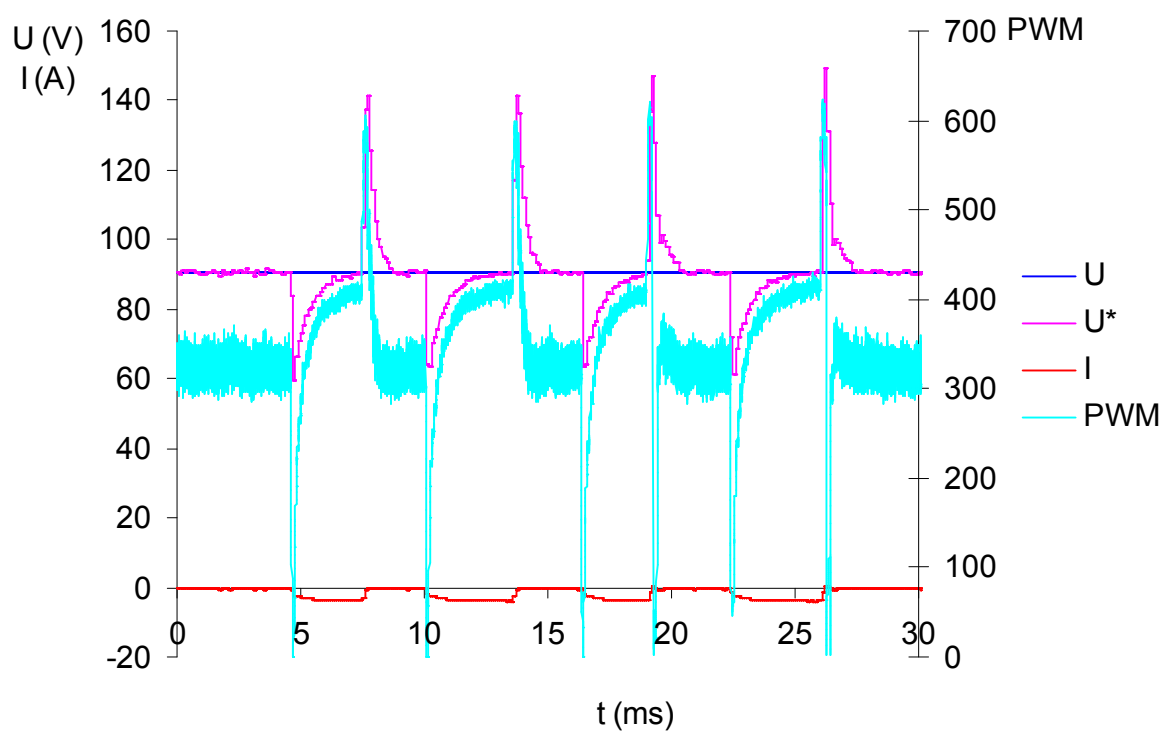


Obr. 7.6 – Regulace otáček se zatžením



Obr. 7.7 – Regulace otáček se změnou zatžení

7.4 Regulace napětí



Obr. 7.8 – Regulace napětí (vliv poruch – změnou zatížení v meziobvodu)

ZÁVĚR

Cílem diplomové práce bylo kompletně zrealizovat řídicí jednotku. Jednalo se tedy jak o hardwarovou realizaci tak i softwarovou. Toto se podařilo splnit. Výsledná realizace nejenže splňuje všechny zadané požadavky, ale navíc byly přidány další funkce. Jedná se o regulaci napětí meziobvodu a řízení dvoukvadrantového měniče (včetně ovládání reverzace s použitím externích stykačů). Jednotka byla navržena, zkonstruována a naprogramována tak, aby se dosáhlo co nejvyšší univerzálnosti použití.

Celá řídicí jednotka je uzavřena v plastové krabici, do které je zasazen zobrazovací displej a panel ovládání. Jednotka nepotřebuje žádný externí zdroj napájení, jelikož si potřebné napětí vyrábí sama ze síťového přívodu. Prostřednictvím jednotky lze napájet i všechna připojená čidla. Základní deska (včetně mikrokontroléru) může být kdykoliv z krabice vyjmuta a aplikována v jiných situacích, kdy je toto uložení řídicí jednotky nevhodné.

Softwarová realizace probíhala ve dvou etapách. V první etapě se připravovaly funkce pro zpracovávání dat. Ve druhé etapě došlo k vlastní implementaci regulačního obvodu a vytvoření kompletní podoby výsledného softwaru. Regulační obvod byl navrhován s ohledem na možnosti použití vybraného mikrokontroléru (počet PWM kanálů) a zvoleného režimu spínání prvků měniče (unipolární).

Řídicí jednotka je schopna pracovat samostatně nebo s připojeným počítačem. Parametrizací jednotky se dosahuje vysoké univerzálnosti jejího použití. Lze tedy regulovat proud, otáčky a napětí v různých aplikacích. Pomocí připojeného počítače je možné stahovat z jednotky hodnoty několika veličin a ty si pak nechat vykreslit. Součástí vybavení jednotky jsou softwarové ochrany (např. nadproud, vysoké otáčky atd.).

Výsledky testovacích měření řídicí jednotky ukazují, že vytvořená regulační struktura vyhovuje. Vhodnějším nastavením parametrů regulátorů lze samozřejmě dosáhnout mnohem lepších charakteristik.

V současné době se předpokládá, že dojde k vytvoření duplikátů a všechny regulační smyčky budou využity pro řešené granty (soustrojí se synchronními a asynchronními motory, elektrokolo) na Katedře elektrotechniky, elektroniky a zabezpečovací techniky v dopravě (KEEZ).

POUŽITÁ LITERATURA

- [1] JAVŮREK, Jiří. Regulace moderních elektrických pohonů. 1. vyd. Praha: Grada Publishing a.s., 2003. 264 s. ISBN 80-247-0507-9.
- [2] Informace o sběrnici RS232 [online]. [cit. 2008-05-20]. Dostupné z WWW: <<http://rs232.hw.cz/>>.
- [3] Informace o IRC čidlu firmy LARM a.s. [online]. [cit. 2008-05-20]. Dostupné z WWW: <<http://www.larm.cz/>>.
- [4] Informace o snímačích od firmy LEM [online]. [cit. 2008-05-20]. Dostupné z WWW: <<http://www.lem.com/>>.
- [5] Dokumentace k mikrokontroléru Atmega64 [online]. [cit. 2008-05-20]. Dostupné z WWW: <<http://www.atmel.com>>.
- [6] Dokumentace k ostatním součástkám [online]. [cit. 2008-05-20]. Dostupné z WWW: <<http://www.datasheetcatalog.com/>>.
- [7] HAASZ, Vladimír, SEDLÁČEK, Miloš. Elektrická měření – Přístroje a metody. 2. vyd. Praha: vydavatelství ČVUT, 2003. 337 s. ISBN 80-01-02731-7
- [8] FOLTÝN, Michal. Řídící jednotka pro ovládání modelového kolejiště: bakalářská práce. Pardubice: Univerzita Pardubice – Dopravní fakulta Jana Pernera, 2006. 45 s., 8 příl.
- [9] MAŠEK, Zdeněk. Přednášky z předmětu Aplikace mikroprocesorů. Pardubice: Univerzita Pardubice – Dopravní fakulta Jana Pernera, 2007.

SEZNAM TABULEK

Tab. 4.1 – Instrukční sada.....	39
Tab. 4.2 – Význam jednotlivých bitů z Tab. 4.1	39
Tab. 4.3 – Komunikační protokol.....	40
Tab. 5.1 – Princip výpočtu regulační odchylky u proudové smyčky	44
Tab. 7.1 – Parametry regulátorů	53

SEZNAM OBRÁZKŮ

<i>Obr. 1.1 – Pulzní měniče: a) čtyřkvadrantový, b) dvoukvadrantový</i>	11
<i>Obr. 1.2 – Průběhy obou typů spínání: a) unipolární, b) bipolární</i>	12
<i>Obr. 1.3 – Činnost pulzního měniče v jednotl. kvadrantech: režimy a), c) mot., b), d) gen.</i>	12
<i>Obr. 1.4 – Napěťové úrovně RS232</i>	14
<i>Obr. 1.5 – 3-vodičové propojení: a) nekřížené, b) křížené</i>	14
<i>Obr. 1.6 – Vzor snímacího kotouče</i>	15
<i>Obr. 1.7 – Průběhy signálů z IRC čidla</i>	15
<i>Obr. 1.8 – Hallův jev</i>	16
<i>Obr. 2.1 – Blokové schéma řídicí jednotky</i>	17
<i>Obr. 2.2 – Blokové schéma LCD</i>	20
<i>Obr. 2.3 – Obvod pro převod signálů A a B na směr a CLK</i>	21
<i>Obr. 2.4 – Průběh signálů z IRC čidla a D-KO</i>	21
<i>Obr. 2.5 – Schéma obvodu napájení</i>	22
<i>Obr. 2.6 – Schéma obvodu zpracování signálů z LEMů</i>	23
<i>Obr. 2.7 – Schéma obvodu zpracování signálů z IRC čidla</i>	24
<i>Obr. 2.8 – Schéma obvodů zapojení displeje (vlevo) a ovládacího modulu (vpravo)</i>	25
<i>Obr. 2.9 – Provedení řídicí jednotky</i>	27
<i>Obr. 3.1 – Fast PWM</i>	30
<i>Obr. 4.1 – Komunikace s displejem</i>	37
<i>Obr. 4.2 – Znaková sada</i>	38
<i>Obr. 5.1 – Regulační obvod</i>	43
<i>Obr. 5.2 – PWM modulátor a výstupní logika</i>	47
<i>Obr. 6.1 – Vývojová struktura hlavní smyčky programu</i>	49
<i>Obr. 6.2 – Vývojová struktura činnosti programu v přerušení</i>	51
<i>Obr. 7.1 – Schéma zapojení testovacího pracoviště</i>	52
<i>Obr. 7.2 – Testovací pracoviště</i>	53
<i>Obr. 7.3 – Řízený stejnosměrný motor (asynchronní motor jako zátěž)</i>	53
<i>Obr. 7.4 – Regulace proudu (při konstantních otáčkách)</i>	54
<i>Obr. 7.5. – Regulace proudu se změnou zatížení</i>	54
<i>Obr. 7.6 – Regulace otáček se zatížením</i>	55
<i>Obr. 7.7 – Regulace otáček se změnou zatížení</i>	55
<i>Obr. 7.8 – Regulace napětí (vliv poruch – změnou zatížení v meziobvodu)</i>	56

SEZNAM ZKRATEK

EEPROM	Electrically-Erasable Programmable Read-Only Memory
SRAM	Static Random Access Memory
JTAG	Joint Test Action Group
CGRAM	Character Generator Random Access Memory
DDRAM	Display Data Random Access Memory
LCD	Liquid Crystal Display
SMD	Surface Mount Device
UART	Universal Asynchronous serial Receiver nad Transmitter
PWM	Pulse Width Modulation
PI	proporcionálně integrační (regulátor)
PS	proporcionálně sumační (regulátor)

SEZNAM PŘÍLOH

PŘÍLOHA č. 1 – Schéma zapojení

PŘÍLOHA č. 2 – Rozmístění konektorů

PŘÍLOHA č. 3a, b – Zapojení vývodů konektorů

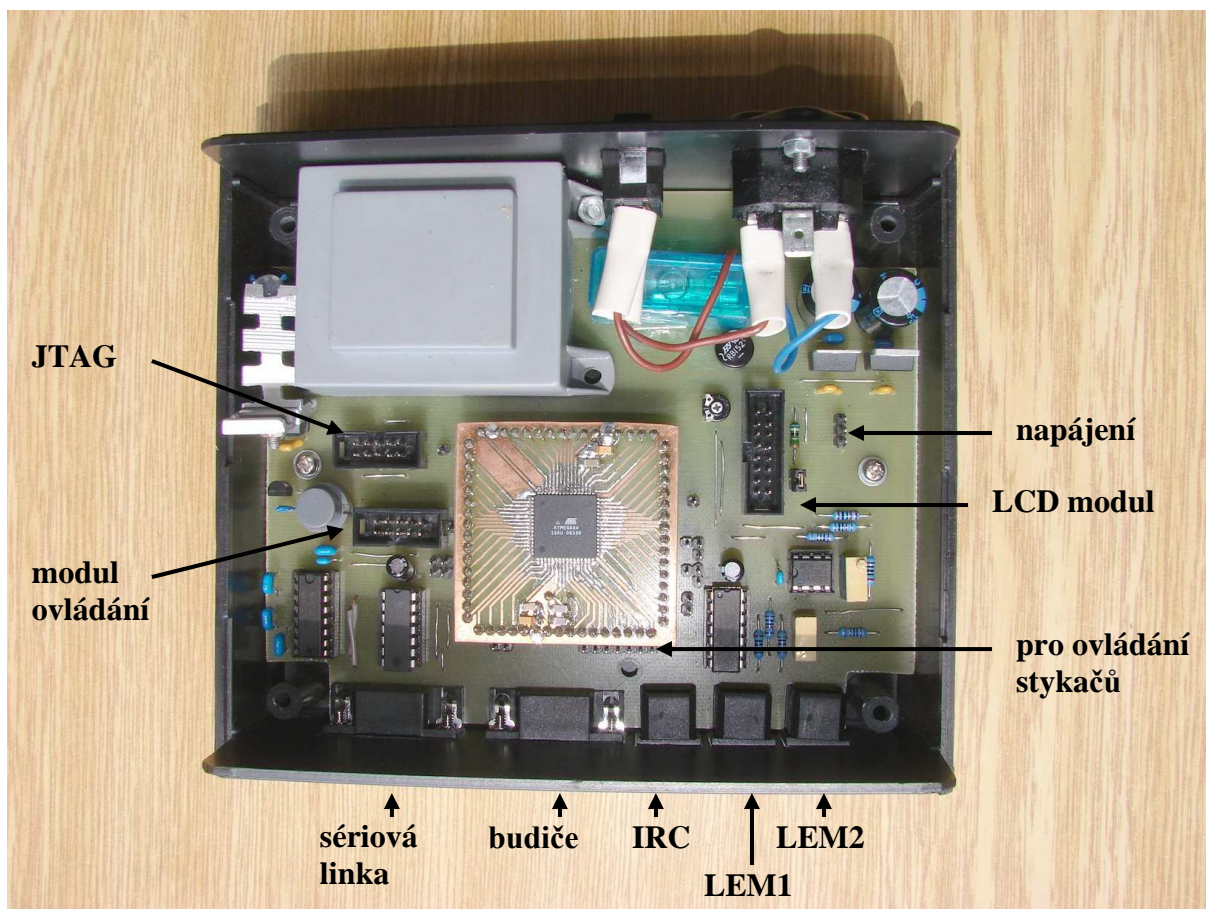
PŘÍLOHA č. 4a, b – Dodatky komunikačního protokolu

PŘÍLOHA č. 5a, b – Testovací měření

PŘÍLOHA č. 6 – Ukázka softwaru v počítači

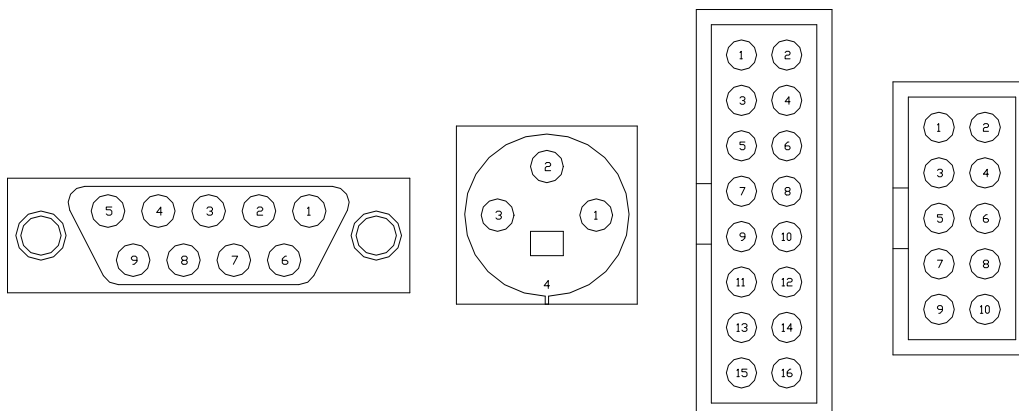
PŘÍLOHA č. 7 – CD se softwarem pro řídicí jednotku a schémata pro její konstrukci

PŘÍLOHA č. 2 – Rozmístění konektorů



PŘÍLOHA č. 3a – Zapojení vývodů konektorů

Označení vývodů konektoru CAN9-F, MINI-DIN3, MLW16 a MLW10



Konektory CAN9-F (přední panel)

Sériová linka		
Význam	Číslo	Význam
-	1 2	TXD
RXD	3 4	-
GND	5 6	
-	7 8	-
-	9	-

Budiče		
Význam	Číslo	Význam
CHB	1 2	V4
V3	3 4	V2
V1	5 6	GND
GND	7 8	GND
GND	9	

Konektory MINI-DIN3 (přední panel)

IRC čidlo		
Význam	Číslo	Význam
GND	1 2	A
B	3 4	GND

LEM1 - snímání napětí		
Význam	Číslo	Význam
-15 V	1 2	U
+15 V	3 4	GND

LEM2 - snímání proudu		
Význam	Číslo	Význam
-15 V	1 2	I
+15 V	3 4	GND

Konektor MLW16 (uvnitř jednotky)

LCD modul		
Význam	Číslo	Význam
GND	1 2	+5 V
V0 (kontrast)	3 4	RS
RW	5 6	E
DB0	7 8	DB1
DB2	9 10	DB3
DB4	11 12	DB5
DB6	13 14	DB7
A (podsvětlení)	15 16	K (podsvětlení)

PŘÍLOHA č. 3b – Zapojení vývodů konektorů (pokračování)

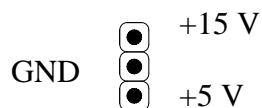
Konektory MLW10 (uvnitř jednotky)

Modul ovládání			
Význam	Číslo		Význam
-	1	2	-
GND	3	4	REG
+5 V	5	6	L/P
START	7	8	STOP
-	9	10	-

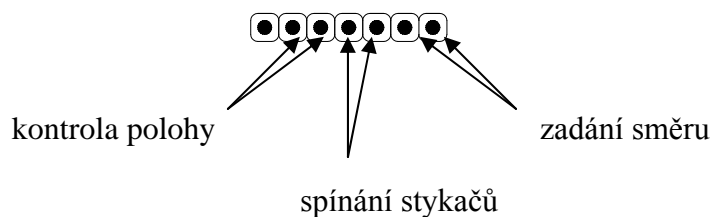
JTAG			
Význam	Číslo		Význam
TCK	1	2	GND
TDO	3	4	+5 V
TMS	5	6	RST
+5 V	7	8	-
TDI	9	10	GND

Lámací lišty (uvnitř jednotky)

- vyvedené piny napájení



- dodatečné vyvedení pinů k ovládání reverzace pro řízení ve dvoukvadrantovém režimu



PŘÍLOHA č. 4a – Dodatky komunikačního protokolu

Seznam parametrů řídicí jednotky (seřazeno podle pořadí zadávání)

- proporcionální zesílení proudového PS regulátoru
- integrační časová konstanta proudového PS regulátoru
- proporcionální zesílení ot./nap. PS regulátoru
- integrační časová konstanta ot./nap. PS regulátoru
- váha filtru pro proudové čidlo
- volba regulované veličiny (proud, otáčky, napětí)
- ovládání (místní, z počítače)
- režim (čtyř-, dvou- kvadrantový)
- rozsah proudu pro místní ovládání
- rozsah otáček pro místní ovládání
- rozsah napětí pro místní ovládání
- mezní proud, překročením dojde k zastavení regulace
- mezní otáčky, překročením dojde k zastavení regulace
- mezní napětí, překročením dojde k zastavení regulace
- váha filtru pro napěťové čidlo
- počet pulsů IRC
- perioda odesílání dat (násobek 2ms)
- číslo první odesílané hodnoty
- číslo druhé odesílané hodnoty
- převodová hodnota proudového čidla
- převodová hodnota napěťového čidla
- číslo třetí odesílané hodnoty
- číslo čtvrté odesílané hodnoty

PŘÍLOHA č. 4b – Dodatky komunikačního protokolu

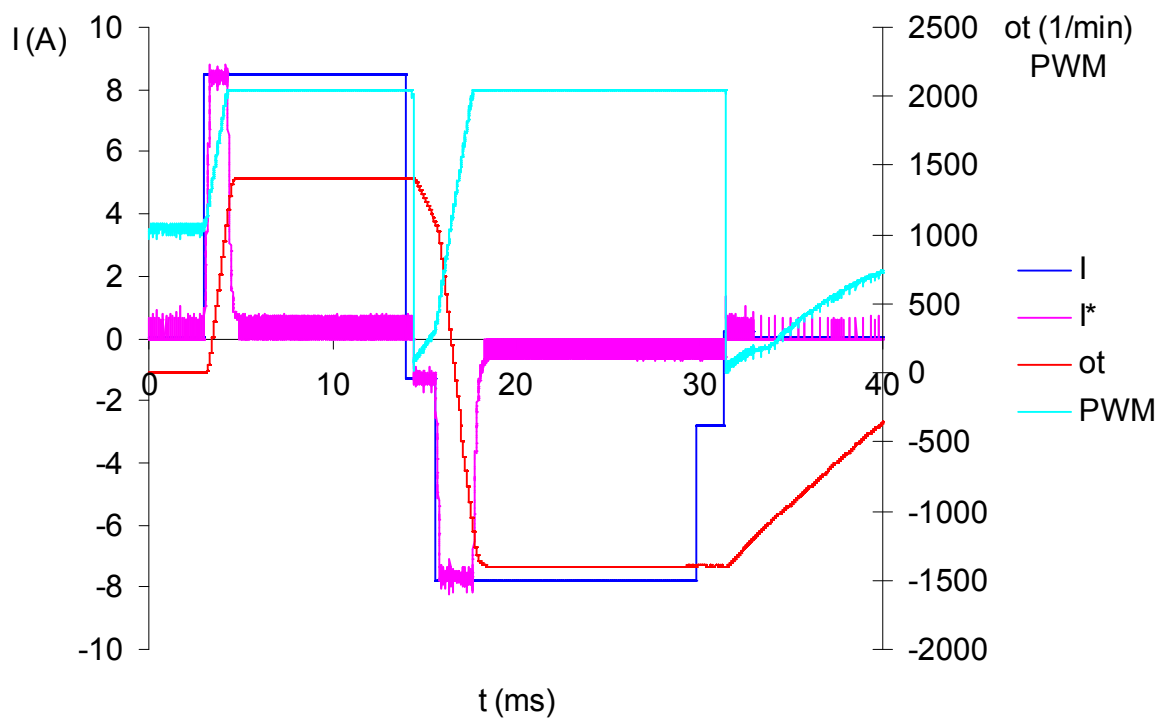
Význam chybových kódů

kód	význam
1	velký proud
2	velké otáčky
3	velké napětí
4	chyba budičů
5	chyba komunikace
6	chyba reverzace (pouze 2Q)

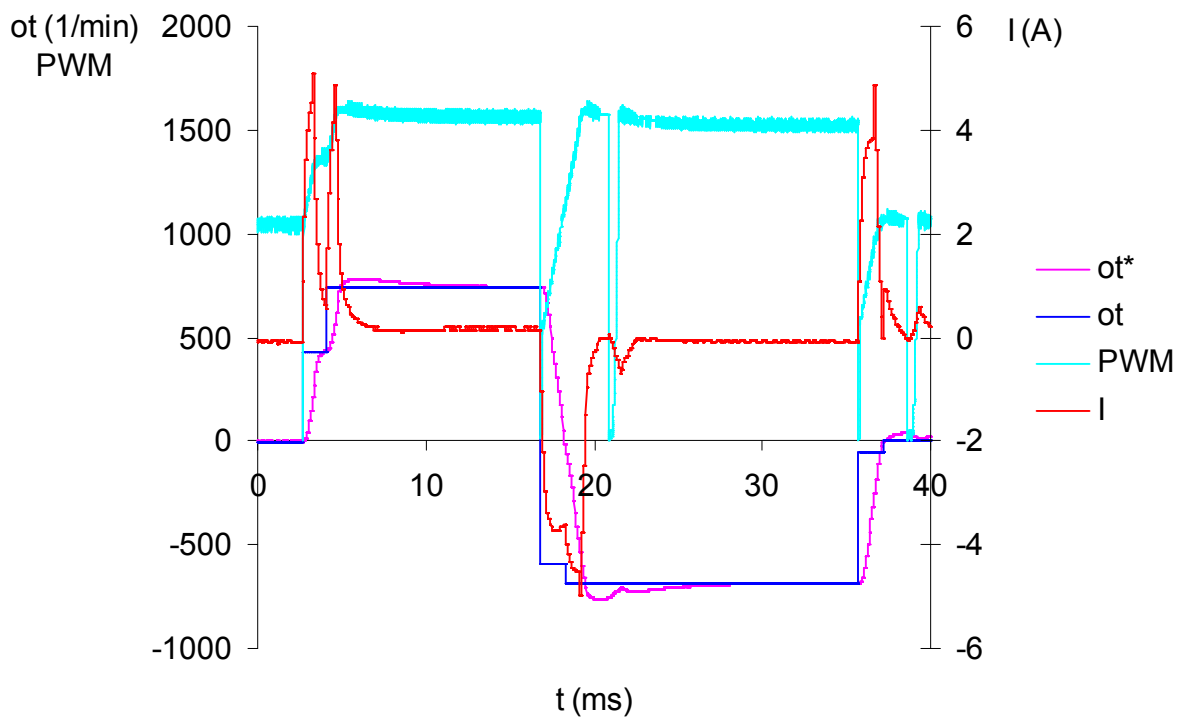
Kódy možných odesílaných hodnot do počítače

kód	význam
1	žádaný proud
2	skutečný proud
3	žádané otáčky
4	skutečné otáčky
5	žádané napětí
6	skutečné napětí
7	hodnota pro PWM modulátor
8	reg. odchylka v proud. smyčce

PŘÍLOHA č. 5a – Testovací měření

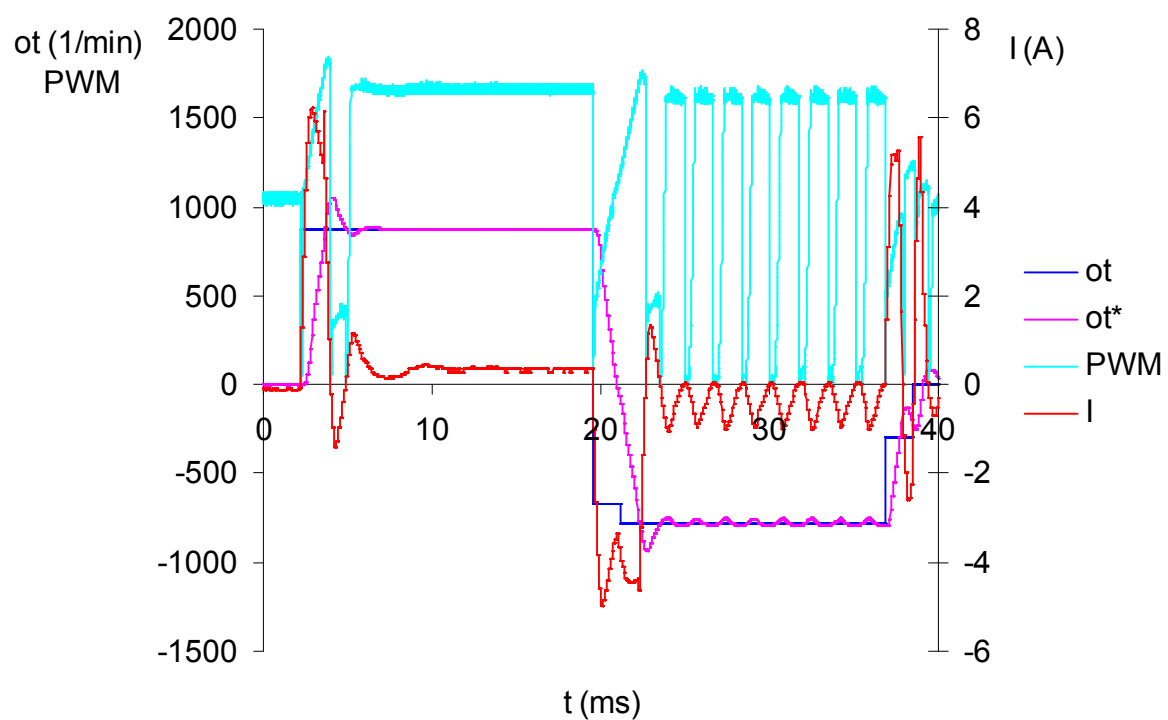


Regulace proudu bez zatížení



Regulace otáček bez zatížení

PŘÍLOHA č. 5b – Testovací měření



Regulace otáček bez zatížení (nevhodně nastavené parametry regulátoru, kmitání při záporné polaritě otáček vlivem offsetu proudového čidla)

PŘÍLOHA č. 6 – Ukázka softwaru v počítači

The screenshot shows the 'Chopper' software interface in the 'Parametrizace' (Parameterization) tab. The window title is 'Chopper' and the menu bar includes 'Soubor' and 'Komunikace'. The interface is divided into several sections for configuring motor parameters:

- Mód regulace:** Otláčeková smyčka (dropdown)
- Typ měniče:** Čtyřkvadrantový (dropdown)
- Ovládání:** po sériovém portu (dropdown)
- Counters:** L_n: 7000, U_n: 200, N_n: 1800, L_max: 400, U_max: 400, N_max: 3000
- Proudový regulátor:** Kp: 2600, Ki: 5
- Otláčekový / napěťový regulátor:** Kp: 250, Ki: 3000
- Váha filtru I:** 50
- Váha filtru U:** 64
- Počítání:** Počet pulsů IPC čidla: 1024, Převod proudového čidla: 12500, Převod napěťového čidla: 200
- Nastavení logování dat:** Perioda odesílání: 2, Proměnná 1: Žádané otáčky, Proměnná 2: Skutečné otáčky, Proměnná 3: Žádaný proud, Proměnná 4: Akční veličina

At the bottom, the status bar shows 'Komunikace OK', 'klid', and '199 93'.

The screenshot shows the 'Chopper' software interface in the 'Řízení' (Control) tab. The window title is 'Chopper' and the menu bar includes 'Soubor' and 'Komunikace'. The interface features control buttons and a speed display:

- Control Buttons:** Start, Stop, Reverzace
- Speed Display:** Žádaná hodnota otáček (Requested speed) with a slider ranging from -1800 ot/min to 1800 ot/min. The current value is -162 ot/min.
- Data Management:** Loguj, Zobraz data, Ulož data buttons

At the bottom, the status bar shows 'Komunikace OK', 'klid', and '199 85'.