



SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE  
Fakulta elektrotechniky a informatiky

# RIADENIE KROKOVÉHO MOTORA MIKROPROCÉSOM

Bakalárska práca

Evidenčné číslo: FEI-5402-26710

Študijný program: Priemyselná informatika  
Pracovisko: Ústav riadenia a priemyselnej informatiky  
Vedúci záverečnej práce/školiťel': Ing. Jozef Dúbravský  
Konzultant:

**Bratislava 2010**

**Alexander Tichý**



## **ANOTÁCIA BAKALÁRSKEJ PRÁCE**

Slovenská technická univerzita v Bratislave  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Študijný program: Priemyselná informatika

Autor: Alexander Tichý

Bakalárska práca: Riadenie krokového motora mikroprocesorom

Vedúci bakalárskej práce: Ing. Jozef Dúbravský

Mesiac, rok odovzdania Máj, 2010

V tejto bakalárskej práci je popísaný návrh a realizácia riadiaceho systému pre unipolárne riadenie krokových motorov. Navrhnutý systém riadi krokový motor pomocou štvortaktného riadenia. Nadriadený PC komunikuje s riadiacou elektronikou pomocou zbernice USB. Súčasťou práce je podrobný opis a postup návrhu jednotlivých častí riadiacej elektroniky, oboznamuje so základnými vlastnosťami a spôsobmi riadenia krokových motorov. Ďalšou časťou je zoznámenie sa s mikrokontrolérom AVR a popisom nastavení jeho integrovaných súčastí.

## **ANNOTATION OF BACHELOR THESIS**

Slovak University of Technology in Bratislava  
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION  
TECHNOLOGY

Course: Industrial Informatics

Author: Alexander Tichý

Bachelor Thesis: Control stepper motor with micro-processor

Supervisor: Ing. Jozef Dúbravský

Month, Year: May, 2010

This bachelor thesis describes the design and implementation of management system for the management of unipolar stepping motors. The proposed system controls stepper motor with four stroke management. Superior PC communicates with the control electronics through the USB bus. The work is a detailed description of the process and draft the various parts of the control electronics, introduces the basic features and ways to control stepping motors. Another part is to get familiarized with the AVR micro controller and a description of its integrated set of components.

## Čestné prehlásenie

Prehlasujem, že som bakalársku prácu vypracoval samostatne s využitím uvedenej odbornej literatúry.

V Bratislave 14.5.2010

.....

## Obsah

|  |    |
|--|----|
| Úvod .....   | 7  |
| 1.0 Krokové motory .....   | 8  |
| 1.1 Základné pojmy.....  | 8  |
| 1.2 Krokové motory s pasívnym rotorom.....                             | 9  |
| 1.3 Krokové motory s radiálne polarizovaným permanentným magnetom..... | 9  |
| 1.4 Krokové motory s axiálne polarizovaným permanentným magnetom.....  | 9  |
| 1.5 Unipolárne riadenie krokových motorov.....                         | 10 |
| 1.6 Bipolárne riadenie krokových motorov.....                          | 12 |
| 1.7 Mikrokrokovanie.....   | 13 |
| 2.0 Architektúra AVR.....  | 14 |
| 2.1 ATmega8.....   | 15 |
| 2.2 Nastavenie programovateľných I/O portov.....                       | 17 |
| 2.3 Nastavenie taktovania CPU.....                                     | 19 |
| 2.4 Časovače.....  | 20 |
| 2.4.1 Režim CTC.....   | 22 |
| 2.4.2 Rýchly PWM režim.....  | 22 |
| 2.4.3 Fázovo korigovaný PWM režim.....                                 | 23 |
| 2.5 Rozhranie RS232.....   | 25 |
| 2.6 Nastavenie USART.....  | 27 |
| 3.0 Realizácia.....  | 29 |
| 3.1 Menič.....   | 29 |
| 3.2 Zdroj.....   | 33 |
| 3.3 Impulzný regulátor.....  | 33 |
| 3.4 Lineárny regulátor.....  | 35 |
| 3.5 Riadenie .....   | 35 |
| 3.6 USB.....   | 37 |
| 4.0 Riadiaci software.....   | 40 |
| 4.1 Riadiaci program v mikro kontroléry.....                           | 40 |
| 4.2 Ovládací program v PC.....   | 46 |
| 4.3 Návod.....   | 47 |
| Záver.....   | 48 |
| Zoznam použitej literatúry.....  | 49 |

## Úvod

Cieľom tejto bakalárskej práce je navrhnúť a realizovať akčný člen pre unipolárne riadenie krokových motorov. Akčný člen bude ovládaný pomocou nadriadeného PC. Pre riadenie krokových motorov je potrebné navrhnúť, vyrobiť a odtestovať riadiacu elektroniku. Táto elektronika sa bude skladať z niekoľkých blokov.

Výkonová časť, ktorá bude zabezpečovať spínanie fáz motora. Tiež musí zabezpečiť prúdovú ochranu motora a zabrániť preťaženiu spínacích súčiastok. V súčasnosti existuje na trhu veľké množstvo špecializovaných súčiastok určených na riadenie krokových motorov, ale tiež je možné realizovať niektoré časti pomocou diskretných súčiastok. Čím je možné dosiahnuť vyššiu zaťažiteľnosť.

Riadiaca časť bude mať za úlohu generovať riadiace signály pre výkonovú časť a zabezpečovať komunikáciu s nadriadeným PC. V riadiacom bloku bude použitý mikrokontrolér ATmega8. ATmega8 je 8-bitový mikrokontrolér založený na architektúre AVR. Významnou úlohou riadiacej časti bude zabezpečovať komunikáciu s užívateľom pomocou nadriadeného PC. Komunikácia bude prebiehať po dnes už veľmi rozšírenej zbernici USB.

Riadiaci program v PC bude mať za úlohu odovzdávať riadiacej elektronike prostredníctvom príkazov požiadavky užívateľa. Nevhodné alebo nerealizovateľné požiadavky budú odmietnuté. Úlohou riadiaceho programu bude zobrazovať priebeh rýchlosti otáčania, alebo periódy riadiaceho signálu.

Súčasťou tejto bakalárskej práce bude zoznámenie sa s architektúrou AVR, podrobný popis mikrokontrolera ATmega8 spolu s príkladmi využitia jeho integrovaných súčastí v rôznych režimoch. Ďalšou časťou bude zoznámenie sa so základnými vlastnosťami krokových motorov, ich mechanickým vyhotovením a spôsobmi ich riadenia. Ďalej postup návrhu riadiacej elektroniky a vysvetlenie významu jednotlivých jeho častí. Pre riadiacu elektroniku bude navrhnutý a odtestovaný riadiaci software, jeho popis bude tiež súčasťou tejto bakalárskej práce spolu s návodom na obsluhu.

## 1.0 Krokové motory

Krokové motory sú synchronne motory prispôsobené k prevádzke v krokovom režime. U krokových motorov je uhol natočenia hriadele určený počtom riadiacich impulzov. Teda hriadeľ motora sa neotáča súvisle, ale o presne definovaný uhol, krok. Táto vlastnosť sa využíva pri ich riadení, nie je potrebný žiadny snímač polohy, poloha sa dá vypočítať. Aby mohol motor pracovať bez priamej spätnej väzby, snímača polohy, je nutné aby pracoval s dostatočnou momentovou rezervou. Moment motora a presnosť natočenia závisia od mechanického prevedenia motora, ale aj od spôsobu riadenia.

Podľa mechanického prevedenia rozdelíme na:

- motory s pasívnym rotorom, nazývané tiež reluktančné.
- motory s aktívnym rotorom.

### 1.1 Základné pojmy

**Krok:** Je mechanická odozva rotora krokového motora na jeden riadiaci impulz, pri tomto pohybe prejde rotor z jednej magnetickej kľudovej polohy do najbližšej magnetickej kľudovej polohy.

**Veľkosť kroku:**  $\alpha$  je uhol daný konštrukciou a spôsobom riadenia motora, ktorý odpovedá zmene polohy rotora spôsobenej jedným riadiacim impulzom.

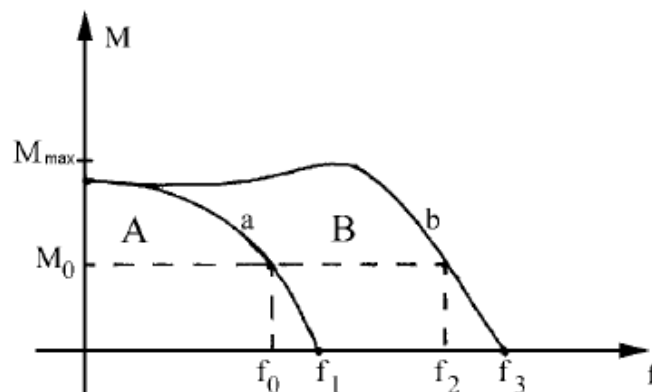
**Magnetická kľudová poloha:** Je poloha, v ktorej sa ustáli rotor nabudeného krokového motora ak sa statický uhol záťaže rovná nule.

**Statický uhol záťaže:**  $\beta$  je uhol, o ktorý sa vychýli rotor nabudeného krokového motora z magnetickej kľudovej polohy pri danej záťaži hriadele motora.

**Statický moment:**  $M_S$  je moment, ktorý je v rovnováhe s krútiacim momentom pôsobiacim na hriadeľ nabudeného krokového motora a vychylujúci rotor o statický uhol záťaže.

**Statická charakteristika krokového motora:** je závislosť statického momentu  $M_S$  na statickom uhle záťaže  $\beta$ .

**Momentová charakteristika krokového motora:** je závislosť momentu  $M$  na frekvencii kroku nabudeného krokového motora, ktorý sa otáča a je zaťažovaný. Krivka b na obrázku č. 1 je pracovná charakteristika krokového motora a krivka a je rozbehová charakteristika krokového motora pri momente zotrvačnosti  $J_z$ . [13]



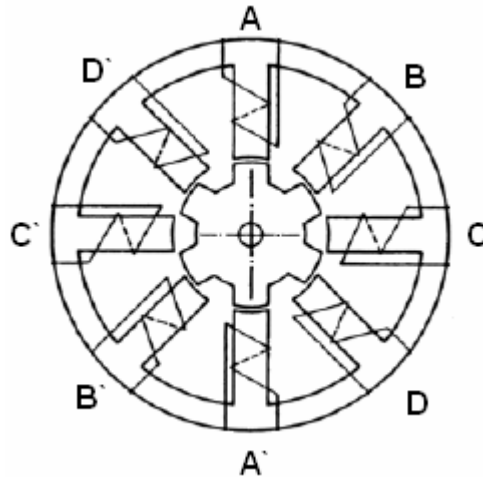
Obrázok č. 1.

Momentová charakteristika krokového motora [13]



## 1.2 Krokové motory s pasívnym rotorom

Stator je tvorený cievkami usporiadanými po obvode motora. Protiľahlé cievky sú elektricky spojené do série a tvoria jednu fázu. Rotor je vyrobený z feromagnetického materiálu a po jeho obvode sú zuby. Počet zubov na rotore je menší ako počet pólov statora. Zjednodušený rez motora je na obrázku č. 2.



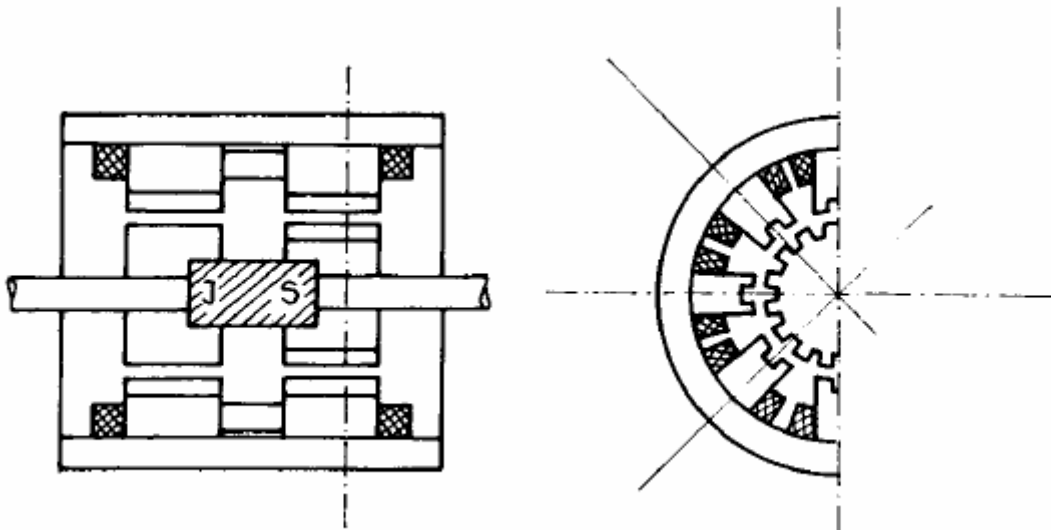
Obrázok č. 2  
Krokový motor s pasívnym rotorom [13]

## 1.3 Krokové motory s radiálne polarizovaným permanentným magnetom

Tieto krokové motory majú podobnú konštrukciu ako krokové motory s pasívnym rotorom. Na rozdiel od krokových motorov s pasívnym rotorom majú motory s aktívnym rotorom v rotore magnet s pólovými nastavcami tvoriacimi zuby rotora. Nastavce sú usporiadané tak, aby sa striedal severný a južný pól. Počet pólov statora je dvojnásobný oproti počtu pólov rotora a zároveň deliteľný štyrmi. Tieto motory so typické veľkým krokom a majú veľkú riadiacu frekvenciu. [13]

## 1.4 Krokové motory s axiálne polarizovaným permanentným magnetom

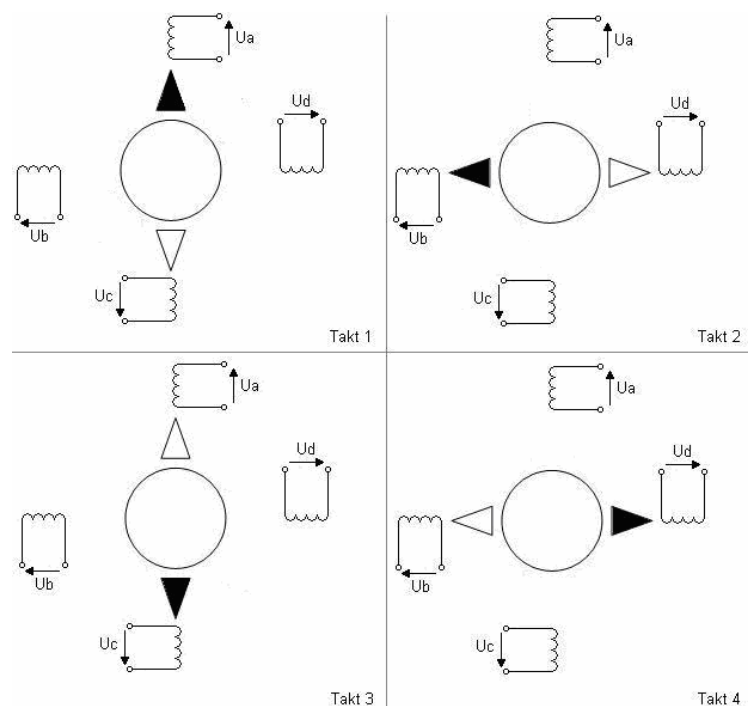
V rotore motora je axiálne polarizovaný permanentný magnet s pólovými nastavcami. Každý z pólových nastavcov má inú magnetickú polarizáciu. Po obvode pólových nastavcov sú zuby. Zuby pólových nastavcov sú vzájomne pootočené o polovicu šírky drážky (rozostup zubov). Počet zubov rotora ovplyvňuje veľkosť kroku (na príklad ak má rotor 50 zubov potom krok je  $1,8^\circ$ ). Stator má po obvode osem cievok podobne ako u krokových motorov s pasívnym rotorom. Na čele cievok sú pólové nastavce so zubami s rovnakou šírkou, ako zuby na rotore. Zjednodušená schéma je na obrázku č. 3.



Obrázok č. 3.  
Schéma krokového motora [13]

### 1.5 Unipolárne riadenie krokových motorov

Základným spôsobom riadenia krokových motorov je postupné pripájanie fáz motora na napätie tak, aby bolo vytvorené rotačné magnetické pole, ktoré vyvolá pohyb rotora. Zjednodušené znázornenie je na obrázku č. 4. Na obrázku č. 5 je časový priebeh fáz v štyroch taktach. Vhodnou zmenou postupnosti fáz vieme meniť zmysel otáčania rotora. Výhodou tohto spôsobu riadenia je nízka spotreba energie, ale motor má tiež najnižší moment. [13]

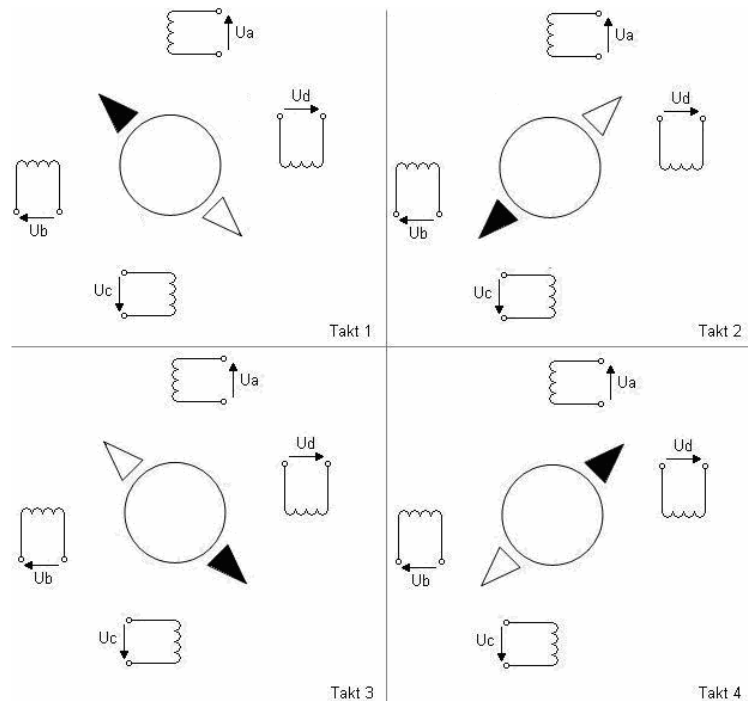


Obrázok č. 4  
Princíp unipolárneho riadenia.

|    |        |        |        |        |
|----|--------|--------|--------|--------|
| Ua | █      |        |        |        |
| Ub |        | █      |        |        |
| Uc |        |        | █      |        |
| Ud |        |        |        | █      |
|    | Takt 1 | Takt 2 | Takt 3 | Takt 4 |

Obrázok č. 5  
Časový priebeh pripájania fáz

Ďalším spôsobom riadenia je tiež štvortaktné riadenie, pri ktorom sú v jednom takte pripojené dve fázy motora. Týmto spôsobom sa dosahuje moment motora  $\sqrt{2}$ -krát väčší ako u predchádzajúceho spôsobu riadenia.

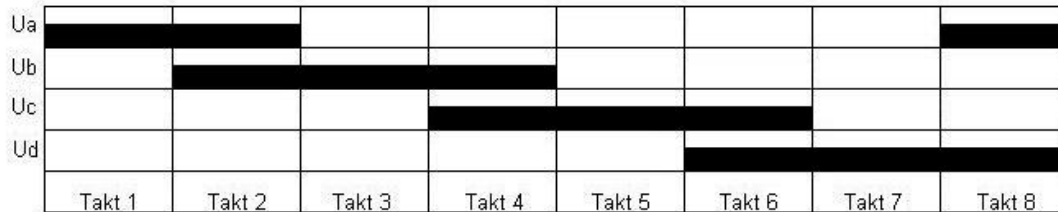


Obrázok č. 6  
Unipolárne riadenie s dvoma aktívnymi fázami

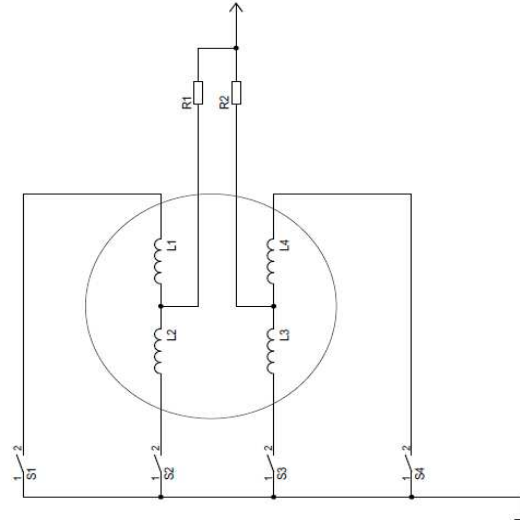
|    |        |        |        |        |
|----|--------|--------|--------|--------|
| Ua | █      | █      |        |        |
| Ub |        | █      | █      |        |
| Uc |        |        | █      | █      |
| Ud | █      |        |        | █      |
|    | Takt 1 | Takt 2 | Takt 3 | Takt 4 |

Obrázok č. 7  
Časový priebeh pripájania fáz s dvoma aktívnymi fázami

Kombináciou týchto spôsobov riadenia je osemtaktné riadenie, ktorým dosiahneme zmenšenie kroku motora o polovicu. Nevýhodou môže byť meniaci sa moment motora v závislosti od polohy rotora.



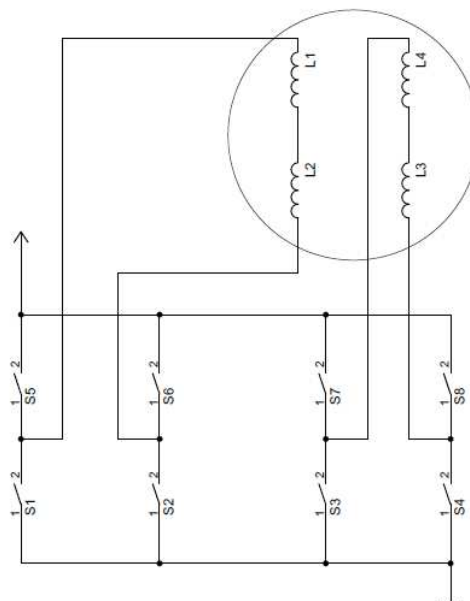
Obrázok č. 8. Časový priebeh osemtaktného riadenia



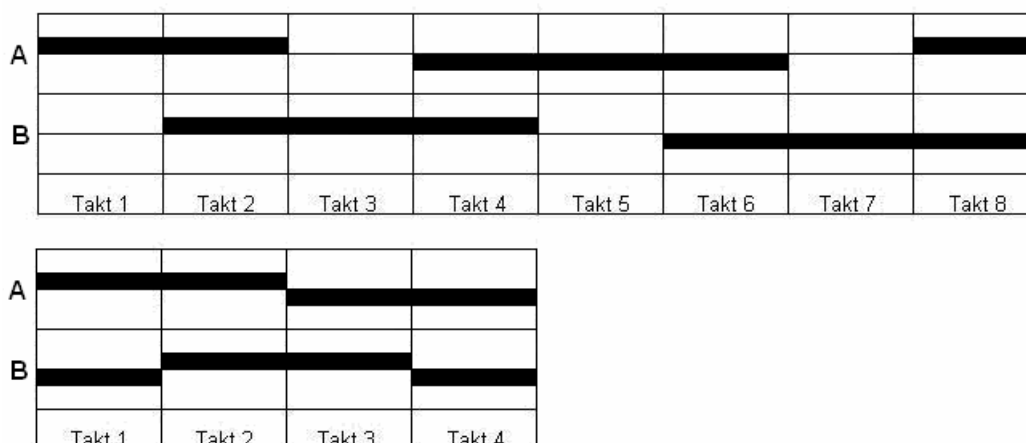
Obrázok č. 9. Principiálna schéma meniča pre unipolárne riadenie

### 1.6 Bipolárne riadenie krokových motorov

Tento spôsob riadenia je náročnejší na riadiacu elektroniku. Na každú fázu motora je potrebný úplný H-mostík obrázok č.10. U tohto spôsobu sa dá tiež realizovať štvor a osem taktné riadenie. Priebeh prúdov je na obrázku č.11.



Obrázok č. 10. Principiálna schéma pre bipolárne riadenie



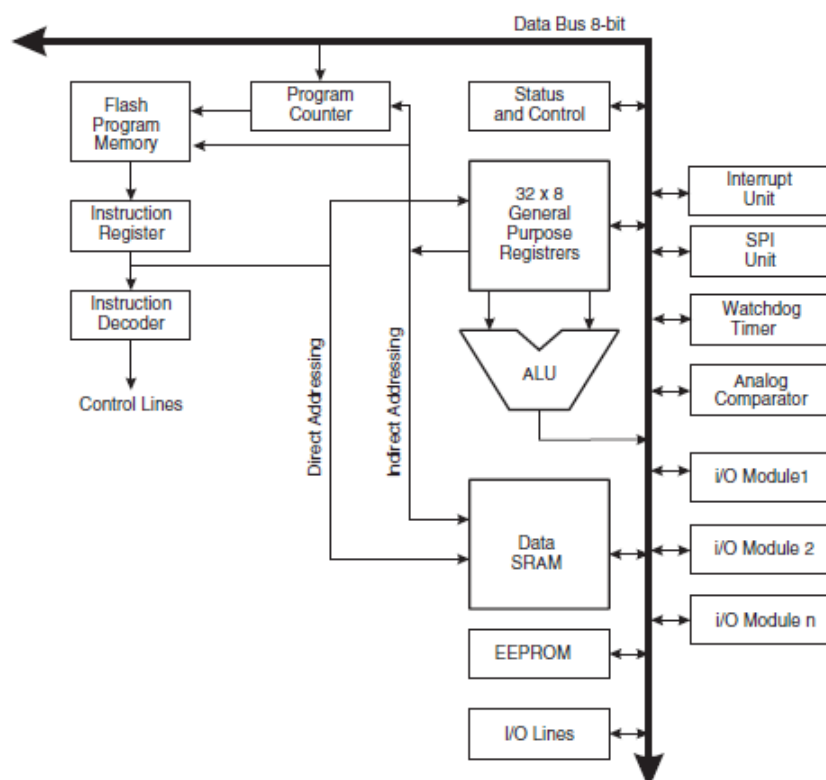
Obrázok č. 11.  
 Časové priebehy osem a štvor taktného, bipolárneho riadenia

### 1.7 Mikrokrokovanie

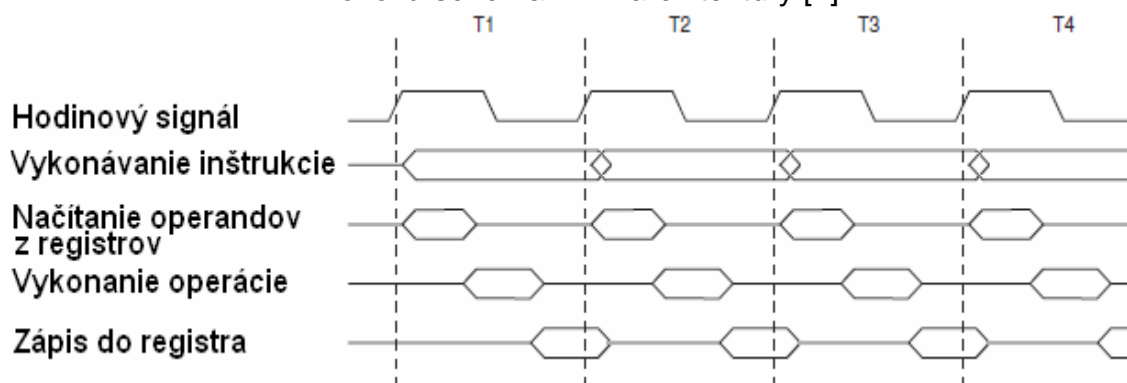
Mikrokrokovanie je technika riadenia, ktorá dokáže ďalej deliť krok motora. V praxi je možné každý krok deliť na 64 až 128 mikrokrokov. Pri klasickom riadení tečie fázami motora rovnaký prúd. Pri mikrokrokovaní sa tento prúd mení a tým sa dá nastaviť ľubovoľná rovnovážna poloha medzi dvoma susednými krokmi. Mikrokrokovanie ďalej zvyšuje nároky na elektroniku.[13]

## 2.0 Architektúra AVR

Architektúru AVR vyvinuli v Nórskom technologickom inštitúte. AVR je architektúra RICS, to znamená, že pracuje s redukovanou inštrukčnou sadou. Procesor má k dispozícii 32 8-bitových registrov. Registre môžu obsahovať dáta aj adresy, nie sú špecializované. Posledných 6 registrov je možné používať vo dvojiciach, čím vzniknú tri 16-bitové registre označované X,Y a Z. Tieto registre sa využívajú na smerníky pri nepriamom adresovaní pamäte dát. Registre sú namapované na začiatku pamäti dát. Prístup do registra sa vykonáva v jednom strojovom cykle. Architektúra AVR je založená na Harvardskej architektúre, teda pamäť programu a pamäť dát sú oddelené. Pamäť programu je prístupná pomocou jedno stupňového pipelining (reťazec inštrukcií). Keď sa jedna inštrukcia vykonáva, nasledujúca inštrukcia je pripravovaná na vykonávanie. [7]



Obrázok č. 12.  
Bloková schéma AVR architektúry [7]



Obrázok č. 13.  
Časový priebeh vykonávania inštrukcií [7]

## 2.1 ATmega8

ATmega8 je 8-bitový mikrokontrolér s jadrom AVR.

Vlastnosti : 8kB Flash pamäti určenej pre program

512B EEPROM

1kB SRAM

32 8-bitových pracovných registrov

Dva 8-bitové časovače

16-bitový časovač

8-kanálový (6-kanálový v prevedení DIL)10-bitový AD-prevodník

Programovateľná sériová linka (USART)

Programovateľný Watchdog

Analógový komparátor

Interný kalibrovaný RC oscilátor na 1,2,4,8 MHz

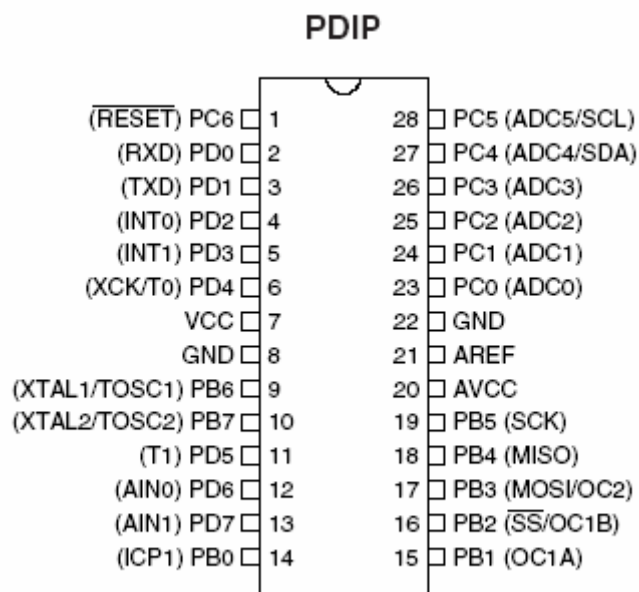
23 programovateľných I/O pinov

Pracovná frekvencia: 0-16MHz pre ATmega8

0-8MHz pre ATmega8L

Napájacie napätie: 4,5-5,5V pre ATmega8

2,7-5,5V pre ATmega8L



Obrázok č. 14.  
Rozloženie pinov v puzdre DIL28 [7]

Popis pinov :

VCC plus pól napájacieho napätia

GND mínus pól napájacieho napätia

PortB (PB0..PB7) je 8-bitový paralelný port

PB7 XTAL2 – pin krištáľového oscilátora 2

TOSC2 – oscilátor časovača

PB6 XTAL1 – pin krištáľového oscilátora 1 alebo vstup externého oscilátora.

TOSC1 – oscilátor časovača

PB5-3 SCK - pin SPI zbernice

MISO – pin SPI zbernice

MOSI – pin SPI zbernice

OC2 – výstup časovača 2

PB2 SS – pin SPI zbernice

OC1B – výstup časovača 1

PB1 OC1A – výstup časovača 1

PB0 ICP1 – vstup časovača 1

PortC (PC0..PC6) je 7-bitový paralelný port

PC6 ak je príznak RSTDISBL=0 potom PC6 sa chová ako Reset

PC5 SCL – pin I<sup>2</sup>C zbernice

ADC5 – analógový vstup do AD – prevodníka kanál 5

PC4 SDA – pin I<sup>2</sup>C zbernice

ADC4 – analógový vstup do AD – prevodníka kanál 4

PC3-0 ADC3-0 – analógový vstup do AD – prevodníka kanál 3-0

PortB (PD0..PD7) je 8-bitový paralelný port

PD7 AIN1 – negatívny vstup analógového komparátora

PD6 AIN0 – pozitívny vstup analógového komparátora

PD5 T1 – taktovací signál časovača 1

PD4 XCK – externé hodiny zbernice USART

T0 – taktovací signál časovača 0

PD3 INT1 – vstup externého prerušenia 1

PD2 INT0 – vstup externého prerušenia 0

PD1 TXD – výstup zbernice USART

PD0 RXD – vstup zbernice USART

Základne elektrické vlastnosti:

Maximálny prúd na pin 40mA

Maximálny napájací prúd 300mA

Maximálne napätie I/O pinu pri log 0 ( $V_{CC}=5V$ ,  $I_{OL}=20mA$ ) je 0,7V

Minimálne napätie I/O pinu pri log 1 ( $V_{CC}=5V$ ,  $I_{OH}=-20mA$ ) je 4,2V

Prúd  $I_{IL}$  pri ( $V_{CC}=5.5V$ ) je 1 $\mu$ A

Prúd  $I_{IH}$  pri ( $V_{CC}=5.5V$ ) je 1 $\mu$ A

$I_{CC}$  v činnosti pri 8MHz,  $V_{CC}=5,5V$  je 15mA

$I_{CC}$  v klude pri 8MHz,  $V_{CC}=5,5V$  je 7mA

Operačná teplota -40C až 85C



## 2.2 Nastavenie programovateľných I/O portov

Vlastnosti jednotlivých portov sa nastavujú pomocou registrov DDRx , POTRBx a bit PUD v registri SFIOR. Čítanie zabezpečuje register PINx (x – označuje jednotlivé porty B, C, D). Vlastnosti vstupu alebo výstupu sa nastavujú pomocou registra DDRx , ak je príslušný bit registra nastavený pin (bit) portu x je výstupný a jeho hodnota je nastavená v registre PORTx na príslušnom bite. Bit PUD obsluhuje funkciu pull-up pre všetky porty naraz. Ak je PUD nastavené na log. 0 funkcia je prístupná. Jednotlivé piny portu, pre ktoré sa má pull-up nastaviť, sa vyberajú pomocou registra PORTx nastavením príslušných bitov registra.

Príklad programu, ktorý nastaví piny 3 a 5 portu B ako výstupné, pričom na pine 3 je log 1 a na pine 5 log 0.

```
void main(void)
{
unsigned char i;

PORTB=(1<<3);           //pin 3 nastaví na log 1
DDRB=(1<<3)|(1<<5);     //pini 3 a 5 nastaví ako výstupné
i=PINB                  //do premennej i načíta vstup z portu B

}
```

Konfiguračné slová ATmega8

Tabuľka č. 1. Lock Byte [7]

| Bit | Názov          | Nastavenie výrobcom | význam   |
|-----|----------------|---------------------|--|
| 5,4 | BLB12<br>BLB11 | 11                  | Zámok Boot loadra<br>11 – odomknuté<br>10 – zákaz zápisu do Boot loadra<br>01 – aplikácia nesmie čítať Boot loader<br>00 – platia obidva zákazy              |
| 3,2 | BLB02<br>BLB01 | 11                  | Zámok programu aplikácie<br>11 – odomknuté<br>10 – zákaz zápisu do oblasti aplikácie<br>01 – Boot loader nesmie čítať aplikáciu<br>00 – platia obidva zákazy |
| 1,0 | LB2<br>LB1     | 11                  | Zámok pamäti<br>11 – pamäť je odomknutá<br>10 – zamknutá pamäť programu aj EEPROM<br>00 – je zamknuté všetko vrátane konfigurácie                            |

Tabuľka č. 2. Low Fuse byte [7]

| Bit         | Názov                                | Nastavenie výrobcom | Význam   |
|-------------|--------------------------------------|---------------------|--|
| 7           | BODLEVEL                             | 1                   | Napätová úroveň podpätia<br>0 – reset pri 2,4 až 2,9 V<br>1 – reset pri 3,7 až 4,5 V |
| 6           | BODEN                                | 1                   | Detekcia podpätia<br>0 – zapnuté<br>1 – vypnuté                                      |
| 5,4         | SUT0<br>SUT1                         | 10                  | Oneskorenie po štarte  |
| 3,2,<br>1,0 | CSSEL0<br>CKSEL1<br>CKSEL2<br>CKSEL3 | 0001                | Voľba zdroja hodinového signálu  |

Tabuľka č. 3. High Fuse byte [7]

| Bit | Názov              | Nastavenie výrobcom | Význam   |
|-----|--------------------|---------------------|--|
| 7   | RSTDISBL           | 1                   | Vypnutie funkcie reset na PC6<br>0 – PC6 je I/O pin<br>1 – PC6 je vstup RESET  |
| 6   | WDTON              | 1                   | Watch Dog Timer<br>0 - Watch Dog Timer je trvale zapnutý<br>1 - Watch Dog Timer je možné vypnúť alebo zapnúť pomocou WDTCR   |
| 5   | SPIEN              | 0                   | 0 – sériové programovanie povolené<br>1 – sériové programovanie zakázané   |
| 4   | CKOPT              | 1                   | Režim oscilátoru<br>0 – výkonový výstup (pre rýchle kryštály)<br>1 – malý rozkmit signálu (menšia spotreba)  |
| 3   | EESAVE             | 1                   | Uchovávanie EEPROM pri mazaní $\mu\text{C}$<br>0 – obsah EEPROM sa zachová<br>1 – obsah EEPROM sa nezachová  |
| 2,1 | BOOTSZ0<br>BOOTSZ1 | 00                  | Veľkosť pamäte vyhradenej pre Boot loader<br>11 – Boot oblasť 128B (do adresy 0xF80)<br>10 – Boot oblasť 256B (do adresy 0xF00)<br>01 – Boot oblasť 512B (do adresy 0xE00)<br>00 – Boot oblasť 1024B (do adresy 0xC00) |
| 0   | BOOTRST            | 1                   | 0 – Po resete sa spustí Boot Loader podľa BOOTSZ0..1<br>1 – Po resete sa spustí program na adrese 0  |

## 2.3 Nastavenie taktovania CPU

Ako zdroj hodinového signálu je možné použiť jeden z piatich zdrojov. Príslušný zdroj sa vyberá nastavením bitov CKSEL0..3, ktoré patria medzi fuse bity. Ďalej je možné nastaviť oneskorenie po reštarte alebo po pripojení napájacieho napätia. Toto oneskorenie sa nastavuje bitami SUT0..1, ale závisí aj od nastavenia CKSEL0..3.

Tabuľka č. 4. Nastavenie zdroja hodinového signálu [7]

| CKSEL3..0      | Zdroj hodinového signálu                     |
|----------------|--|
| 0000           | Externý signál                               |
| 0001           | Interný RC oscilátor 1MHz                    |
| 0010           | Interný RC oscilátor 2MHz                    |
| 0011           | Interný RC oscilátor 4MHz                    |
| 0100           | Interný RC oscilátor 8MHz                    |
| 0101           | Externý RC oscilátor 0.1 – 0.9MHz            |
| 0110           | Externý RC oscilátor 0.9 – 3 MHz             |
| 0111           | Externý RC oscilátor 3 – 8 MHz               |
| 1000           | Externý RC oscilátor 8 – 12 MHz              |
| 1001           | Externý nízko-frekvenčný křišťál (32.768kHz) |
| 101x           | Low power XTAL 0.4 – 0.9 MHz pre CKOPT=1     |
| 110x           | Low power XTAL 0.9 – 3 MHz pre CKOPT=1       |
| 111x           | Low power XTAL 3 – 8 MHz pre CKOPT=1         |
| 101x,110x,111x | XTAL nad 1MHz pre CKOPT = 0                  |

Tabuľka č. 5. Nastavenie oneskorenia pri použití křišťálového oscilátora [7]

| CKSELO | SUT1..0 | Štartovací čas v taktovacích impulzoch | Oneskorenie po reštarte pri Ucc=5V | Doporučené použitie                                      |
|--------|---------|--|------------------------------------|--|
| 0      | 00      | 258                                    | 4,1ms                              | keramický rezonátor, Rýchly nábeh odberu                 |
| 0      | 01      | 258                                    | 65ms                               | keramický rezonátor, Pomalý nábeh odberu                 |
| 0      | 10      | 1024                                   | -                                  | keramický rezonátor, povolenie detekcie poklesu napätia  |
| 0      | 11      | 1024                                   | 4,1ms                              | keramický rezonátor, Rýchly nábeh odberu                 |
| 1      | 00      | 1024                                   | 65ms                               | keramický rezonátor, Pomalý nábeh odberu                 |
| 1      | 01      | 16384                                  | -                                  | Křišťálový oscilátor, povolenie detekcie poklesu napätia |
| 1      | 10      | 16384                                  | 4,1ms                              | Křišťálový oscilátor, Rýchly nábeh odberu                |
| 1      | 11      | 16384                                  | 65ms                               | Křišťálový oscilátor, Pomalý nábeh odberu                |

Nastavenie z výroby je interný RC oscilátor s frekvenciou 1MHz.

## 2.4 Časovače

ATmega8 je vybavený dvoma 8-bitovými časovačmi a jedným 16-bitovým.

Časovač 0 je 8-bitový časovač. Bitmi 0,1,2 (CS00,CS01,CS02) v registre TCCR0 sa nastavuje delič hodinového signálu, alebo externý zdroj z pinu T0. Register TCNT0 slúži ako počítadlo. Prerušenie sa nastavuje v registre TIMSK bit TOIE0. Prerušenie nastane po pretečení registra TCNT0.

Tabuľka č. 6. Nastavenie TCCR0 [7]

| CS2..0 | Význam  |
|--------|---|
| 000    | Časovač je zastavený                          |
| 001    | hodinový signál / 1                           |
| 010    | hodinový signál / 8                           |
| 011    | hodinový signál / 64                          |
| 100    | hodinový signál / 256                         |
| 101    | hodinový signál / 1024                        |
| 110    | Externý signál T0 (reaguje na dobežnej hrane) |
| 111    | Externý signál T0 (reaguje na nábežnej hrane) |

Časovač 1 je 16-bitový časovač. Režimy časovača sa nastavujú v registroch TCCR1A, TCCR1B. Ďalej sa v týchto registroch nastavuje delič a módy výstupov OC1A , OC1B. Register TCNT1H a TCNT1L slúžia ako počítadlá. Register OCR1AH, OCR1AL, OCR1BH a OCR1BL slúžia ako registre komparátorov A a B. Register ICR1H, ICR1L menia svoj význam v závislosti od režimu časovača. Je možné ich použiť na ukladanie hodnoty časovača pri vzniku vonkajšej udalosti, alebo do nich uložiť hodnotu, pri ktorej sa bude časovač nulovať.

Prerušenia sa nastavujú v registri TIMSK bity:

TOIE1 – prerušenie nastane po pretečení počítadla.

OCIE1B – prerušenie spúšťané komparátorom B

OCIE1A – prerušenie spúšťané komparátorom A

TICIE1 – prerušenie spúšťané udalosťou na pine ICP1

Tabuľka č. 7. Nastavenie deliča [7]

| CS2..0 | Význam   |
|--------|--|
| 000    | Časovač je zastavený                           |
| 001    | hodinový signál / 1                            |
| 010    | hodinový signál / 8                            |
| 011    | hodinový signál / 64                           |
| 100    | hodinový signál / 256                          |
| 101    | hodinový signál / 1024                         |
| 110    | hodinový signál T0 (reaguje na dobežnej hrane) |
| 111    | hodinový signál T0 (reaguje na nábežnej hrane) |

Režimy časovača sa nastavujú pomocou bitov WGM1(3..0) v registroch TCCR1A a TCCR1B.

Tabuľka č. 8. Nastavenie režimov časovača 1 [7]

| režim | WGM1 (3..0) | Význam                             | Maximum | Začiatková hodnota OCR1x | Vznik prerušenia |
|-------|-------------|------------------------------------|---------|--------------------------|------------------|
| 0     | 0000        | Normálny režim                     | 0xFFFF  | Aktuálna                 | Max              |
| 1     | 0001        | Fázovo korigovaný PWM              | 0x00FF  | Max                      | Dno              |
| 2     | 0010        | Fázovo korigovaný PWM              | 0x01FF  | Max                      | Dno              |
| 3     | 0011        | Fázovo korigovaný PWM              | 0x03FF  | Max                      | Dno              |
| 4     | 0100        | CTC                                | OCR1A   | Aktuálna                 | Max              |
| 5     | 0101        | Rýchle PWM                         | 0x00FF  | 0x00                     | Top              |
| 6     | 0110        | Rýchle PWM                         | 0x01FF  | 0x00                     | Top              |
| 7     | 0111        | Rýchle PWM                         | 0x03FF  | 0x00                     | Top              |
| 8     | 1000        | Fázovo a frekvenčne korigovaný PWM | ICR1    | 0x00                     | Dno              |
| 9     | 1001        | Fázovo a frekvenčne korigovaný PWM | OCR1A   | 0x00                     | Dno              |
| 10    | 1010        | Fázovo korigovaný PWM              | ICR1    | Max                      | Dno              |
| 11    | 1011        | Fázovo korigovaný PWM              | OCR1A   | Max                      | Dno              |
| 12    | 1100        | CTC                                | ICR1    | Aktuálna                 | Max              |
| 13    | 1101        | Rezervovaný                        | -       | -                        | -                |
| 14    | 1110        | Rýchle PWM                         | ICR1    | 0x00                     | Top              |
| 15    | 1111        | Rýchle PWM                         | OCR1A   | 0x00                     | Top              |

Časovač 2 je 8-bitový časovač zameraný na PWM a asynchrónne operácie. Pomocou registra TCCR2 je možné voliť režimy časovača, nastavovať delič a módy výstupu OC2. Register TCNT2 slúži ako čítač časovača. OCR2 je register komparátora. ASSR register asynchrónnych stavov. Ak bit AS2 v ASSR je 0 časovač je riadený hodinovým signálom procesora, ak je 1 potom je časovač riadený asynchrónne externým nízkofrekvenčným krištálom. Bity TCN2UB, OCR2UB a TCR2UB a ASSR sú príznaky zápisu do príslušných registrov.

Tabuľka č. 9. Nastavenie deliča frekvencie [7]

| CS2..0 | Význam                |
|--------|-----------------------|
| 000    | Časovač je zastavený  |
| 001    | Vstupný signál / 1    |
| 010    | Vstupný signál / 8    |
| 011    | Vstupný signál / 32   |
| 100    | Vstupný signál / 64   |
| 101    | Vstupný signál / 128  |
| 110    | Vstupný signál / 256  |
| 111    | Vstupný signál / 1024 |

Tabuľka č. 10. Režimy časovača 2 [7]

| režim | WGM2 (1..0) | Význam                | Maximum | Začiatková hodnota OCR1x | Vznik prerušenia |
|-------|-------------|-----------------------|---------|--------------------------|------------------|
| 0     | 00          | Normálny režim        | 0xFF    | Aktuálna                 | Max              |
| 1     | 01          | Fázovo korigovaný PWM | 0xFF    | Max                      | Dno              |
| 2     | 10          | CTC                   | OCR2    | Aktuálna                 | Max              |
| 3     | 11          | Rýchle PWM            | 0xFF    | 0x00                     | Max              |

### 2.4.1 Režim CTC

V tomto režime sa vynuluje register TCNTn po zhode s registrom OCRn. Teda je možné regulovať rozsah časovača a tým generovať impulzy rôznej frekvencie [7].

$$f = \frac{f_{clk}}{2N(1+OCRn)} \quad [\text{Hz}]$$

N je nastavenie deliča frekvencie (8, 32, 64, 128, 256, 1024).

Tab.11. Nastavenie výstupu OCn pre normálny a CTC režim [7]

| Bity COMn(1..0) | význam                  |
|-----------------|-------------------------|
| 00              | Bez výstupu na OCn      |
| 01              | Negovanie OCn           |
| 10              | Vynulovanie OCn         |
| 11              | Nastavenie OCn na log.1 |

### 2.4.2 Rýchly PWM režim

V rýchlom PWM režime je začiatková hodnota registra, TCNTn, 0. register TCNTn je inkrementovaný v hodinovom cykle časovača, po dosiahnutí maxima je znovu register nastavený na hodnotu 0. Na výstupe OCn sa podľa nastavenia mení hodnota raz pri zhode s registrom OCRn a raz pri dosiahnutí maxima. U časovača 1 je možné maximálnu hodnotu registra meniť (Tabuľka č. 8. režimy časovača 1). Na obrázku č. 15 sú znázornené priebehy signálov.

Tabuľka č. 12. Režimy výstupu OCn [7]

| Bity COMn(1..0) | význam   |
|-----------------|--|
| 00              | Bez výstupu na OCn   |
| 01              | -  |
| 10              | Vynulovanie OCn pri zhode a Nastavenie OCn na log.1 pri max. hodnote |
| 11              | Nastavenie OCn na log.1 pri zhode a vynulovanie OCn pri max. hodnote |

Frekvencia PWM signálu je [7]:

$$f_{ocn} = \frac{f_{clk}}{N(1 + \max)} \text{ [Hz]}$$

N je nastavenie deliča frekvencie(8, 32, 64,128, 256, 1024). Max je maximálna nastavená hodnota registra TCNTn.

### 2.4.3 Fázovo korigovaný PWM režim

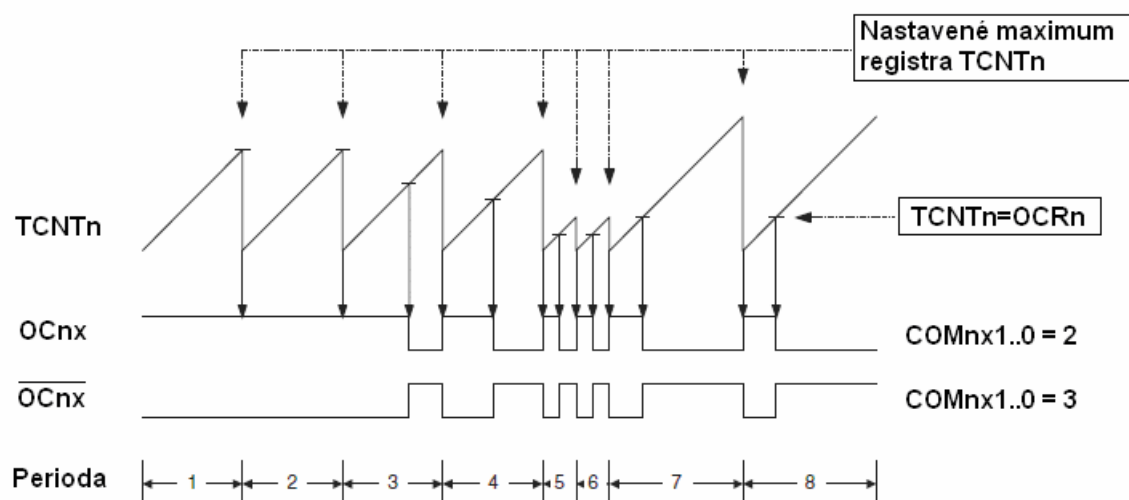
V tomto režime čítač číta od 0 do maxima a potom z maxima do 0. Hodnota na výstupe OCn sa mení vždy pri zhode registrov TCNTn a OCRn podľa nastavenia.

Tabuľka č. 13. Režimy výstupu OC2 [7]

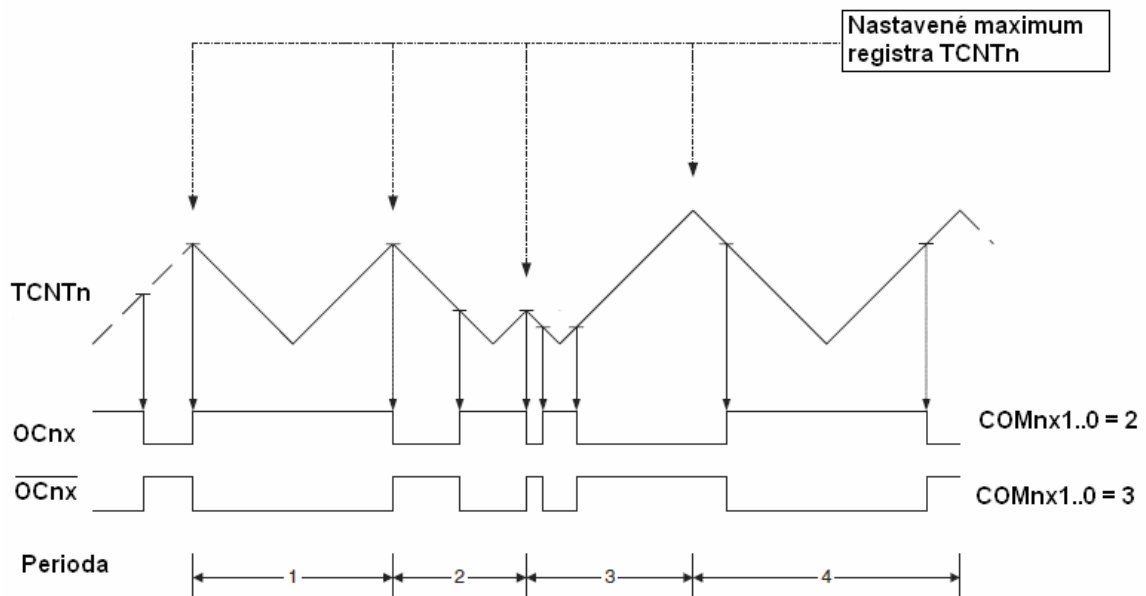
| Bity COMn(1..0) | význam  |
|-----------------|---|
| 00              | Bez výstupu na OCn  |
| 01              | -   |
| 10              | Vynulovanie OCn pri zhode pri čítaní nahor a nastavenie OCn na log.1 pri čítaní smerom dole |
| 11              | Nastavenie OCn na log.1 pri zhode pri čítaní nahor a vynulovanie OCn pri čítaní smerom dolu |

Frekvencia PWM signálu je  $f = \frac{f_{clk}}{2N \times \max}$  [Hz]

N je nastavenie deliča frekvencie(8, 32, 64,128, 256, 1024). Max je maximálna nastavená hodnota registra TCNTn.



Obrázok č. 15. Priebehy signálov v rýchlom PWM režime [7]



Obrázok č. 16.  
Priebehy signálov vo fázovo korigovanom PWM režime [7]

Príklad:

Led dióda pripojená na pin OC2 bliká s plynulou zmenou jasů.

```
#include <mega8.h>
int s=0;
unsigned char k=0;

//obsluha prerušenia časovača 0
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
    if (s==0)
    {
        k++;
        if (k==255)
            s=1;
    }
    if (s==1)
    {
        k--;
        if (k==0)
            s=0;
    }
    OCR2=k;          //nastavenie pomeru PWM (jas Led)
    TCNT0=0xF0;
}
}
```



```

void main(void)
{
  DDRB=0x08;      //nastaví pin PB3 (OC2) ako výstupný
  TCCR0=0x04;     //nastavenie deliča frekvencie pre časovač 0 na
                  //1000000Hz/256=3906,25Hz

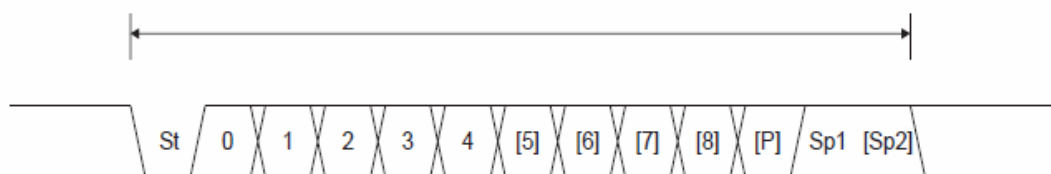
  TCNT0=0xF0;
  TCCR2=0x69;     //nastavenie PWM režimu a delič
                  //frekvencií na 1 pre časovač 2

  TIMSK=0x01;    //povolenie prerušenia pre časovač 0
  #asm("sei")    //globálne povolenie prerušenia
  while (1)
  {
  };
}

```

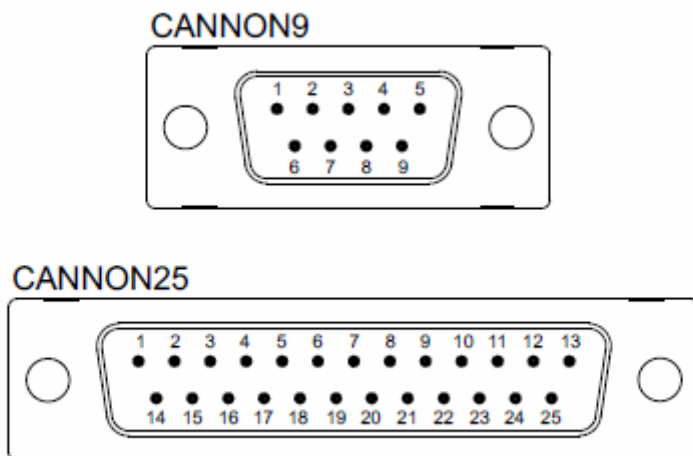
## 2.5 Rozhranie RS232

RS232 je sériová zbernica, ktorá používa dvojlinkové vedenie. Jednu linku na príjem dát (pin RxD), druhú na posielanie (pin TxD). Je možné použiť ďalšiu linku po ktorej master posiela synchronizačné impulzy XCK. Každý poslaný bajt je orámovaný na začiatku štart bitom, ktorý pošle na zbernicu nízku úroveň. Potom sa pošle 0-tý až n-tý bit posieleného bajtu plus paritný bit (ak je nastavený) a nakoniec jeden alebo dva stop bity, podľa nastavenia, ktoré majú vždy vysokú úroveň na zbernici. V kludovom stave zbernice má vysokú úroveň obrázok č. 17. Pre pripojenie ATmega8 k PC je potrebné previesť TTL úrovne na rs232. K tomuto účelu slúži IO MAX232, ktorý obsahuje oddeľovače konvertujúce napäťové úrovne. Napätie potrebné na komunikáciu po RS232 sa vyrába v externých kondenzátoroch.



**St** - štart bit  
**n** - data  
**P** - parita  
**Sp** - stop bit

Obrázok č. 17.  
Rámec jedného bajtu [7]



Obrázok č. 18.  
Konektory používané pre RS232

Tabuľka č. 14. Rozloženie signálov na konektoroch cannon 9 a cannon 25 [9]

| signál | Číslo pinu na cannon 9 | Číslo pinu na Cannon 25 |  |
|--------|------------------------|-------------------------|--|
| CD     | 1                      | 8                       | Modem oznamuje terminálu, že na linke detekuje nosnú frekvenciu        |
| RXD    | 2                      | 3                       | Príjem dát   |
| TXD    | 3                      | 2                       | Vysielanie dát   |
| DTR    | 4                      | 20                      | Terminál týmto signálom oznamuje modemu, že je pripravený komunikovať  |
| GND    | 5                      | 7                       | Zem (u cannon 25 je pin 1 spojený s krytom konektoru)                  |
| DSR    | 6                      | 6                       | Modem týmto signálom oznamuje terminálu, že je pripravený komunikovať  |
| RTS    | 7                      | 4                       | Terminál týmto signálom oznamuje modemu, že komunikačná cesta je voľná |
| CTS    | 8                      | 5                       | Modem týmto signálom oznamuje terminálu, že komunikačná cesta je voľná |
| RI     | 9                      | 22                      | Modem oznamuje terminálu, že na linke detekuje signál zvonenia         |

## 2.6 Nastavenie USART

Je možné nastaviť štyri základné režimy, normálny asynchrónny, asynchrónny režim s dvojnásobnou rýchlosťou, synchrónny režim master a synchrónny režim slave. Bitom UMSEL v registre UCSRC sa nastavuje asynchrónny režim ak UMSEL=0, alebo synchrónny režim. Dvojnásobná rýchlosť asynchrónneho režimu sa nastavuje bitom U2X=1 v registre UCSRA. V synchrónnom režime sa pomocou bitu DDRD4 v registre DDRD nastavuje master ak DDRD4=1 alebo slave režim. Počet stop-bitov sa nastavuje pomocou bitu USBS v registre UCSRC.

Ak je USBS=0 počet vyslaných stop-bitov je 1 ak USBS=1 počet vyslaných stop-bitov je 2. Bitmi UPM1..0 sa nastavuje parita. Parita môže byť párna alebo nepárna. Párna parita znamená, že hodnota paritného bytu sa dopĺňa tak, aby kontrolný súčet log 1. v posielanom rámci bol párný. Teda ak posielame 01001001 potom paritný bit bude 1. U nepárnej parity sa dopĺňa tak, aby bol kontrolný súčet nepárny.

Tabuľka č. 14. Nastavenie parity [7]

| Bity UPM1..0<br>v registri UCSRC | parita         |
|----------------------------------|----------------|
| 00                               | vypnutá        |
| 01                               | -              |
| 10                               | párna parita   |
| 11                               | nepárna parita |

Tabuľka č. 15. Počet prenesených bitov [7]

| UCSZ2..0 z registra<br>UCSRC | Počet bitov |
|------------------------------|-------------|
| 000                          | 5           |
| 001                          | 6           |
| 010                          | 7           |
| 011                          | 8           |
| 100                          | -           |
| 101                          | -           |
| 110                          | -           |
| 111                          | 9           |

Tabuľka č. 16. Prenosová rýchlosť sa nastavuje registrom UBRR podľa vzťahov[7]

| Režim   | Rýchlosť [bit/s]        | Hodnota UBRR<br>v závislosti od prenosovej<br>rýchlosti |
|---|-------------------------|---|
| Normálny<br>asynchrónny režim<br>U2X=0                  | $BAUD=fosc/16*(UBRR+1)$ | $UBRR=(fosc/16*BAUD)-1$                                 |
| Asynchrónny režim<br>s dvojnásobnou<br>rýchlosťou U2X=1 | $BAUD=fosc/8*(UBRR+1)$  | $UBRR=(fosc/8*BAUD)-1$                                  |
| Synchrónny režim<br>Master                              | $BAUD=fosc/2*(UBRR+1)$  | $UBRR=(fosc/2*BAUD)-1$                                  |

Maximálna hodnota UBRR je 4095. Register UDR zabezpečuje čítanie prijatých dát a zápis posielaných dát. Pričom zápis je povolený príznakom UDRE v registre UCSRA.

Príklad: Časovač 2 je v režime CTC jeho rozsah sa dá nastavovať pomocou USARTu.

```
#include <mega8.h>
#include <stdio.h>
```

```
unsigned char i=1;
```

```
// obsluha prerušenia timer 2
interrupt [TIM2_COMP] void timer2_comp_isr(void)
{
    PORTC = i;
    i*= 2;
    if (i == 16)
        i = 1;
}
```

```
//obsluha prerušenia prijímania dát z USART
interrupt [USART_RXC] void príjem_dat(void)
{
    OCR2=getchar();
    putchar(OCR2); //pošle späť potvrdenie
}
```

```
void main(void)
{
    DDRC=0x0F; //piny PC(0..3) sú výstupné
    ASSR=0x08; //zdrojom taktovania časovača je externý křišťál 32.768kHz
    TCCR2=0x0C; //frekvencia externého křišťálu sa delí 64 krát a je nastavený
                //režim CTC

    TCNT2=0x00;
    OCR2=10; //začiatkový "rozsah" čítača
    UCSRA=0x02; //dvojnásobná rýchlosť pre Asynchrónny režim
    UCSRB=0x98; //Povolenie príjmu, povolenie vysielania, povolenie prerušenia
                //pre príjem
    UCSRC=0x86; //Asynchrónny režim, 8bit,1 stop bit, bez parity
    UBRRH=0x00;
    UBRL=0x0C; //rýchlosť 9600 bps chyba 0,2%
```

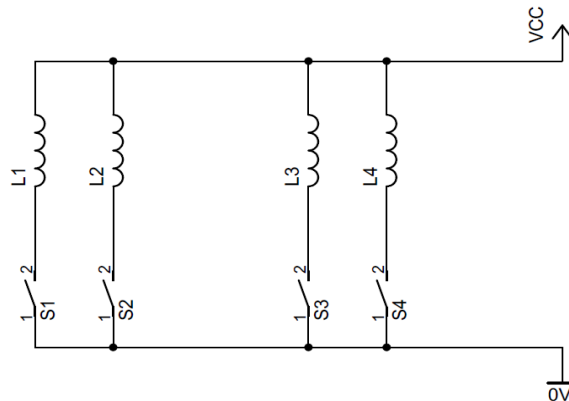
```
TIMSK=0x80; //povolenie prerušenia pre komparátor časovača 2
#asm("sei") //globálne povolenie prerušenia
```

```
while (1)
{
    };
}
```

### 3.0 Realizácia

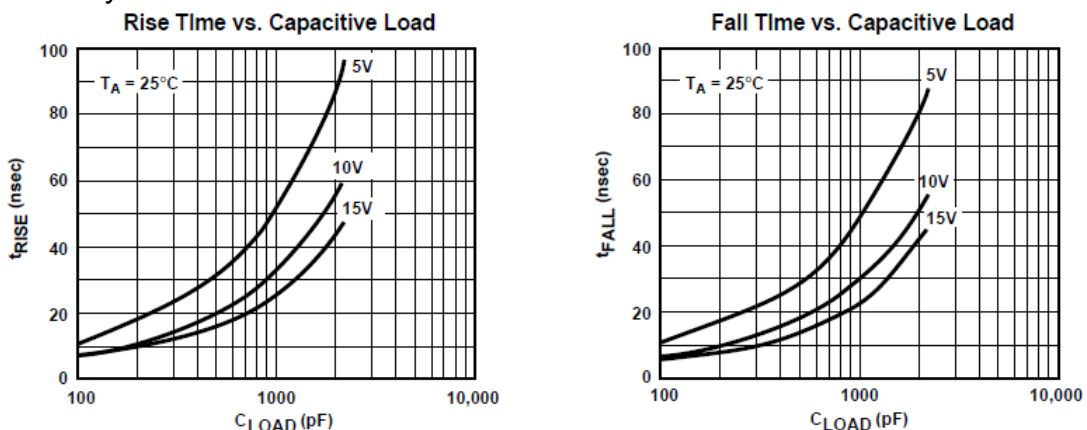
#### 3.1 Menič

Menič slúži na spínanie jednotlivých fáz motora. Na obrázku č. 19. je principiálna schéma meniča pre unipolárne riadenie krokového motora. Pri zopnutí jednotlivých spínačov sa spínajú jednotlivé cievky. Po zopnutí spínača S1 tečie prúd cievkou L1. Spínače je možné realizovať rôznymi výkonovými polovodičmi. Často sa používajú integrované H-mostíky, ktoré sú ale vhodnejšie pre bipolárne riadenie motora.

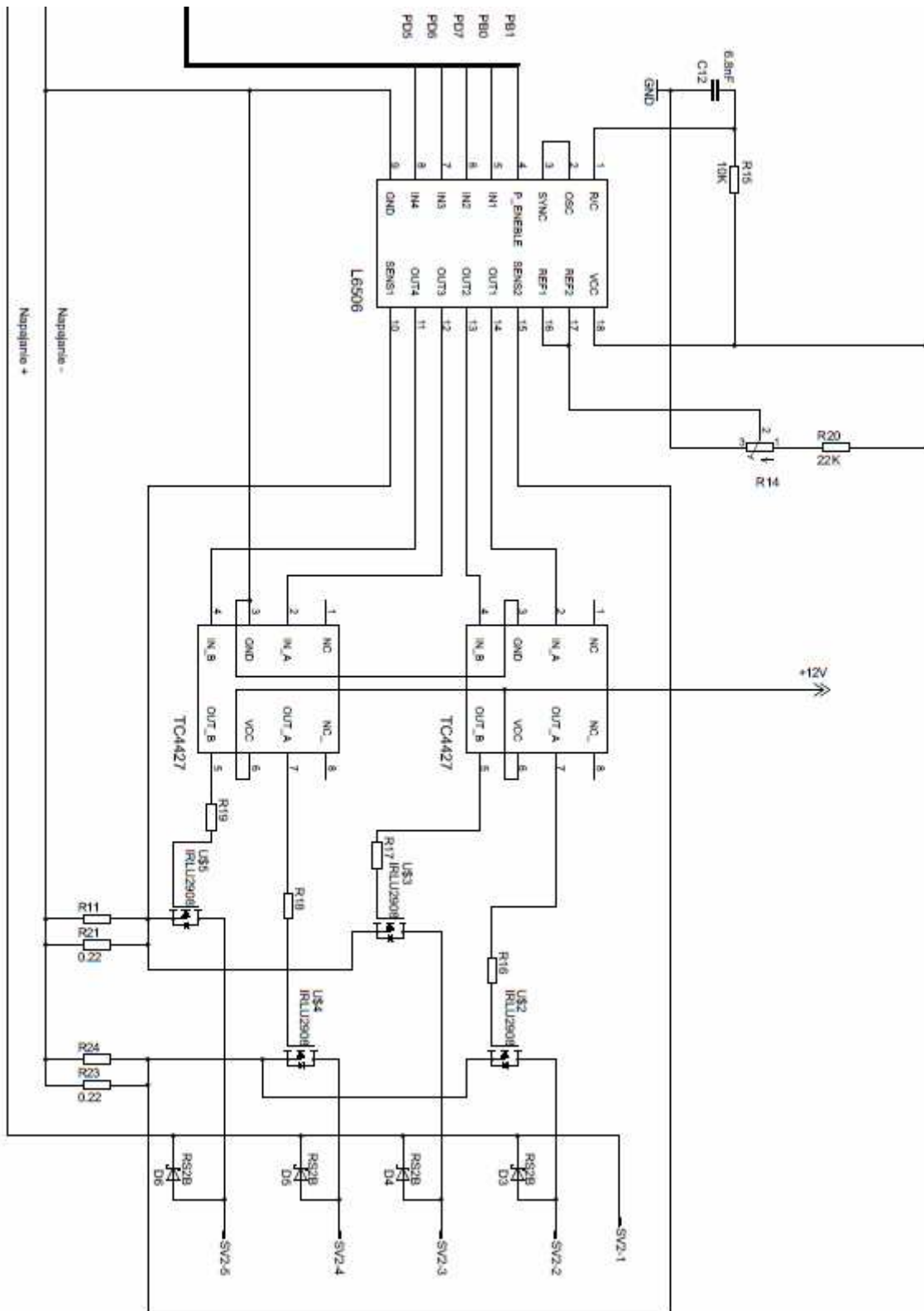


Obrázok č. 19. Schéma ideálneho meniča pre unipolárne riadenie

Na obrázku č. 21 je schéma reálneho meniča použitého v zapojení. Ako spínacie prvky sú použité unipolárne tranzistory IRLU2908. Tieto tranzistory sú typu N-MOSFET s indukovaným kanálom. Maximálne napätie medzi elektródami D a S je  $V_{DSS}=80V$ . Maximálny prúd  $I_D=30A$  a maximálne napätie  $U_{GS}=\pm 16V$ . Ich spínanie je zabezpečené integrovanými obvodmi TC4427. Tento obvod má za úlohu previesť signál z 5V úrovne na signál vhodný na riadenie MOSFET tranzistorov. Takýto signál musí byť schopný dostatočne rýchlo otvoriť tranzistor. To znamená, že musí nabiť respektíve vybiť parazitnú kapacitu  $C_{ISS}$ . U zvoleného typu tranzistora, podľa katalógového listu [1], je táto hodnota 1890pF. Podľa katalógového listu integrovaného obvodu TC4427 [2] trvá nabíjanie pri napájaní 10V približne 55ns vybijanie 50ns, na obrázku č. 20 je graf závislosti nabíjacieho a vybijacieho času od kapacity záťaže. Čím rýchlejšie sa tranzistor otvorí, respektíve zatvorí, tým menšie sú výkonové straty pri prechode medzi otvoreným a zatvoreným stavom tranzistora.

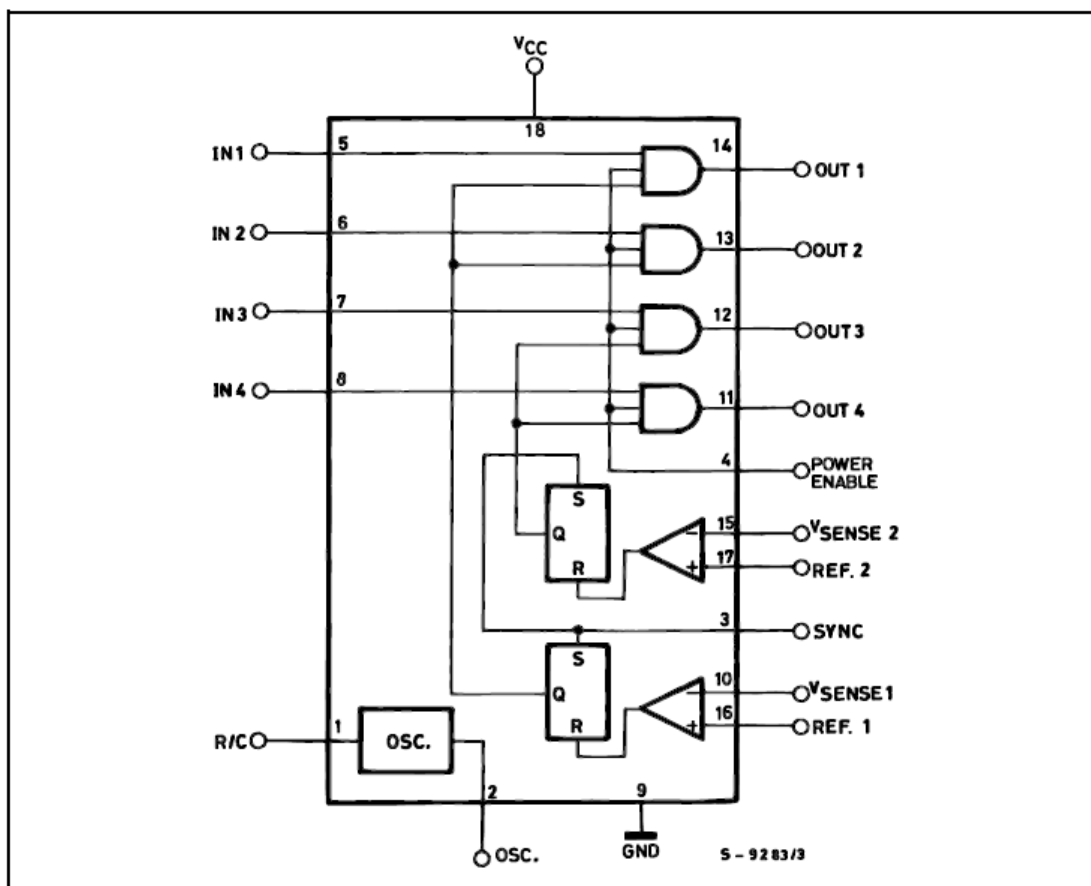


Obrázok č. 20. Nabíjací, vybijací čas v závislosti na  $C_{LOAD}$  [2]



Obrázok č. 21.  
Schéma realizovaného meniča

Integrovaný obvod L6506 zabezpečuje obmedzovanie prúdu tečúceho motorom. Bloková schéma [3] je na obrázku č. 22 .



Obrázok č. 22. Bloková schéma L6506 [3]

Podľa blokovej schémy je riadiaci signál privedený do trojvstupového bloku AND. Teda na to, aby sa signál objavil na výstupe, musia byť aj ostatné dva vstupy nastavené na log 1. Druhý vstup je Power enable. Tento vstup slúži na povolenie činnosti motora, pripojenie na napätie z vonku. Tretí vstup bloku AND je pripojený na výstup Q bistabilného klopného obvodu typu Reset-Set. Výstup takéhoto klopného obvodu sa nastavuje na log 1. vstupom set. A na hodnotu log 0. ho nastaví vstup reset. Pričom na nastavenie stačí dostatočne dlhý impulz. Vstupy reset sú riadené komparátormi. Pozitívny vstup komparátora je pripojený na zdroj referenčného napätia. Negatívny vstup je pripojený na merací rezistor. Vstup set klopného obvodu je vyvedený ako synchronizačný vstup. V použítom zapojení je pripojený na integrovaný oscilátor. Frekvencia oscilátora je nastavená RC členom na pin 1. podľa vzťahu z katalógového listu [3]:

$$F = \frac{1}{0,69 \cdot R15 \cdot C12}$$

Pri použitých hodnotách je frekvencia oscilátora 21,312 kHz. Ak úbytok napätia na meracích rezistoroch stúpne nad referenčné napätie, komparátor resetuje klopný obvod a tým odpojí príslušné vinutie motora. Znovu k pripojeniu vynutia motora príde po príchode impulzu z oscilátora a úbytok napätia na meracích rezistoroch je menší ako referenčné napätia a vstupy IN a power enable sú nastavené na

Log 1. Maximálne referenčné napätie, doporučené výrobcom integrovaného obvodu L6506, je 3V. Z toho vyplýva že maximálny meraný prúd je:

$$I = \frac{U}{R} = \frac{3V}{0,11\Omega} = 27,27A$$

Hodnota R je paralelná kombinácia dvoch 0.22Ω rezistorov (R11 a R21, R23 a R24). Táto hodnota bola zvolená z ohľadom na stratový výkon na meracích rezistoroch. Použité rezistory sú v 7W prevedení. Maximálny prúd je :

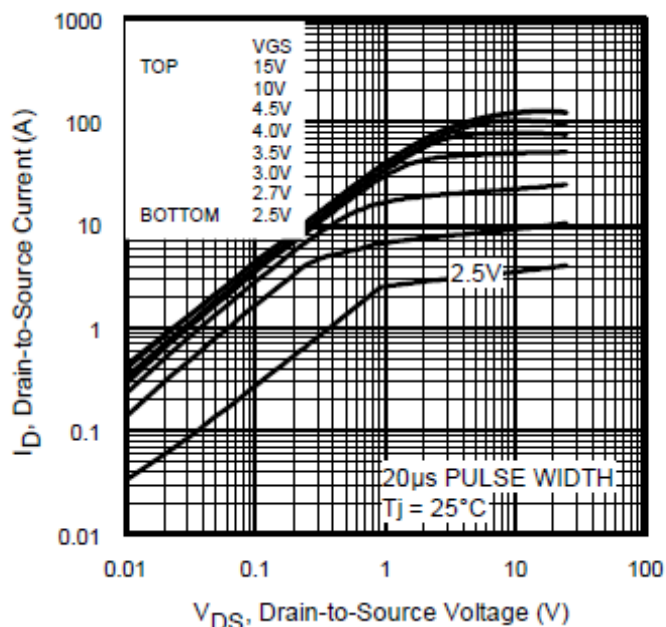
$$I = \sqrt{\frac{P}{R}} = \sqrt{\frac{2 \cdot 7W}{0,11\Omega}} = 11,82A$$

Úbytok napätia na meracích rezistoroch je  $U = R \cdot I = 0,11\Omega \cdot 11,82A = 1,30V$  o túto hodnotu klesne napätie  $U_{GS} = 12V - 0,025V$ . Výsledné napätie  $U_{GS} = 11,975V - 1,30V = 10,675V$ . Podľa katalógového listu tranzistora pri napätí  $U_{GS} = 10V$  je maximálny prúd tečúci tranzistorom 30A. Podľa obrázku č. 23, z katalógového listu IRLU2908 [1], je úbytok na tranzistore pri  $U_{GS} = 10V$  a  $I_D = 12A$  približne 0,3V. A stratový výkon

$$P = U \cdot I = 0,3V \cdot 12A = 3,6W .$$

Teplota tranzistora bez chladenia.

$$T = P \cdot R_{\theta/A} = 3,6W \cdot 40 \frac{^{\circ}C}{W} = 144^{\circ}C$$



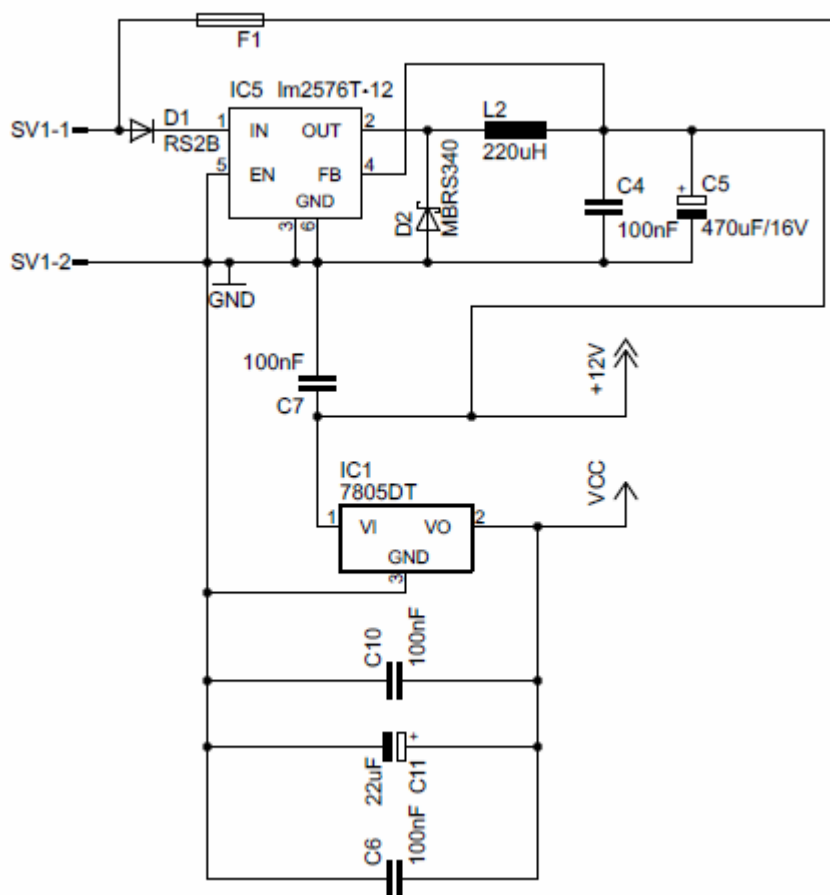
Obrázok č. 23.

Výstupná charakteristika tranzistora IRLU2908 [1]



### 3.2 Zdroj

Zariadenie pracuje s tromi úrovňami napätia. Riadiaca elektronika používa 5V, budiče spínacích tranzistorov sú napájané 12V a motor pracuje priamo so vstupným napájacím napätím, chránený tavnou poistkou.



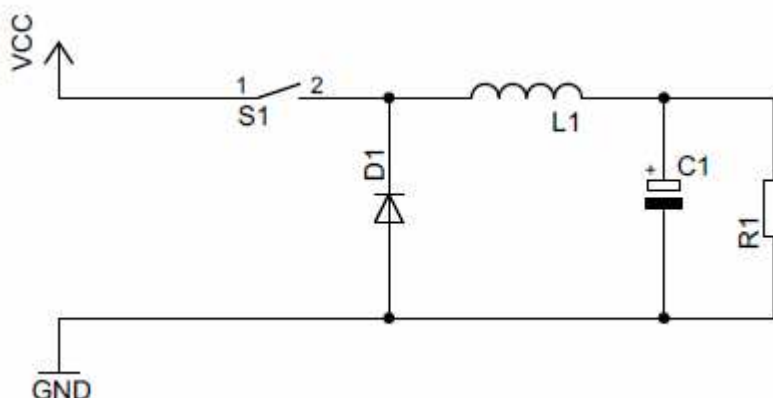
Obrázok č. 24 Schéma zdroja

Vstupné napätie je privádzané cez ochrannú diódu D1 do impulzného regulátora napätia. Tento regulátor je tvorený integrovaným obvodom IC5, diódou D2, cievkou L2 a kondenzátorom C5. Výstupné napätie je 12V, používa sa k napájaniu budičov a ako zdroj pre 5V lineárny stabilizátor tvorený IC1 a filtračným kondenzátorom C11.

### 3.3 Impulzný regulátor

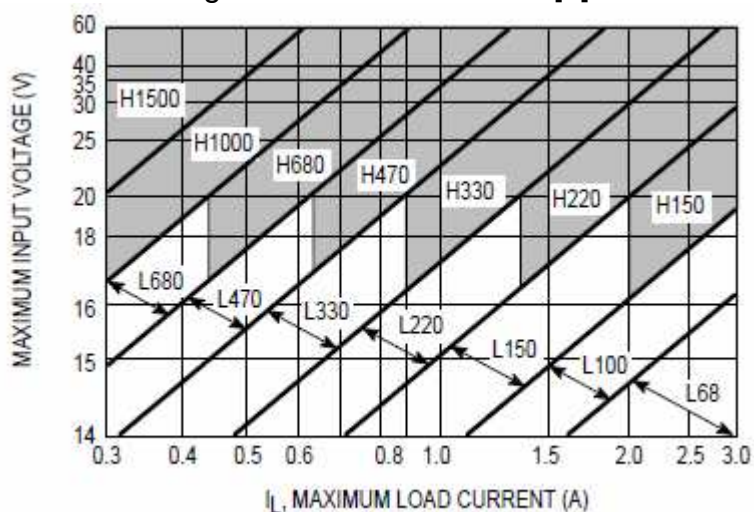
Principiálna schéma impulzného regulátora je na obrázku č. 25. Zdroj sa skladá z výkonového spínacieho prvku, LC filtra a rekuperačnej diódy. Princíp je v cyklickom nabíjaní a vybíjaní cievky L1. Ak je zopnutý spínač S1 začne narastať prúd tečúci cez cievku do záťaže R1 a späť do zdroja. S narastajúcim prúdom začne narastať aj napätie na R1. Ak napätie na záťaži R1 prekročí nastavenú hodnotu riadiaca elektronika rozopne spínač S1 a cievka sa začne chovať ako zdroj. Prúd preteká cievkou do záťaže R1 a cez diódu späť do cievky. Po klesnutí napätia pod nastavenú hodnotu riadiaca elektronika znova zopne spínač S1

a proces sa opakuje. Kondenzátor C1 slúži na zmiernenie zvlnenia napätia spôsobeného nabíjaním a vybíjaním cievky.



Obrázok č. 25.  
Schéma impulzného regulátora

Impulzný regulátor bol zvolený pre vysokú účinnosť a veľký rozsah vstupného napätia oproti lineárnym regulátorom. V zapojení je použitý integrovaný regulátor LM2576-12, ktorý vyžaduje minimum externých súčiastok. Výstupné napätie je pevne nastavené integrovaným deličom na 12V. Indukčnosť cievky L2 je určená podľa obrázku č. 26 z katalógového listu LM2576-12 [4] oblasť L220.



Obrázok č. 26. pracovné oblasti pre cievky [4]

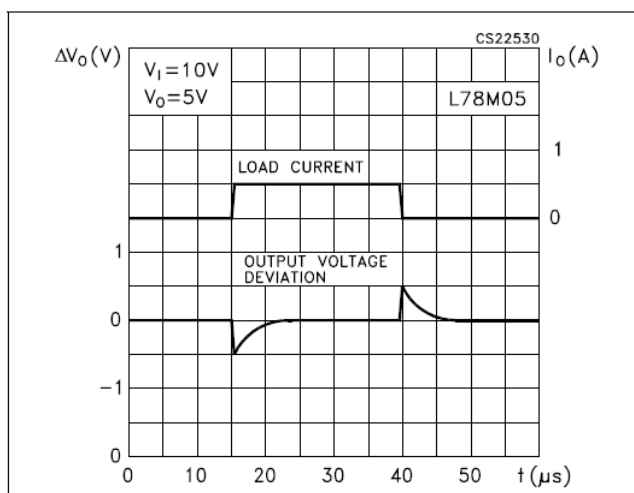
Kapacita kondenzátora C5 je určená vzťahom podľa katalógového listu [4].

$$C_{out} \geq 13300 \cdot \frac{V_{IN(max)}}{V_{out} \cdot L[\mu H]} [\mu F] = 13300 \cdot \frac{40V}{12V \cdot 220\mu H} = 201,5\mu F$$

Teda kapacita kondenzátora C5 musí byť väčšia nanajvýš rovná 201,5μF. Dióda D2 MBRS340 je tiež doporučená výrobcom regulátora. Ide o Schotkyho diódu. Tieto sa používajú pre krátku dobu prepínania medzi vodivým (priepustným) a nevodivým (záverným) stavom. Podľa katalógového listu MBRS340 [5] je maximálne záverné napätie 40V a maximálny prúd 3A.

### 3.4 Lineárny regulátor

Lineárne regulátory regulujú výstupné napätie pomocou zmeny vnútorného odporu. Ich nevýhodou sú veľké výkonové straty spôsobené princípom regulácie, ale ich výhodou je krátka doba regulácie. V zapojení je použitý integrovaný regulátor L78M05. Tento obvod má pevne nastavené výstupné napätie na 5V, maximálny prúd je 0,5A. Podľa katalógového listu [6] je doba regulácie pri skokovom pripojení záťaže 8 $\mu$ s obrázok č. 27.



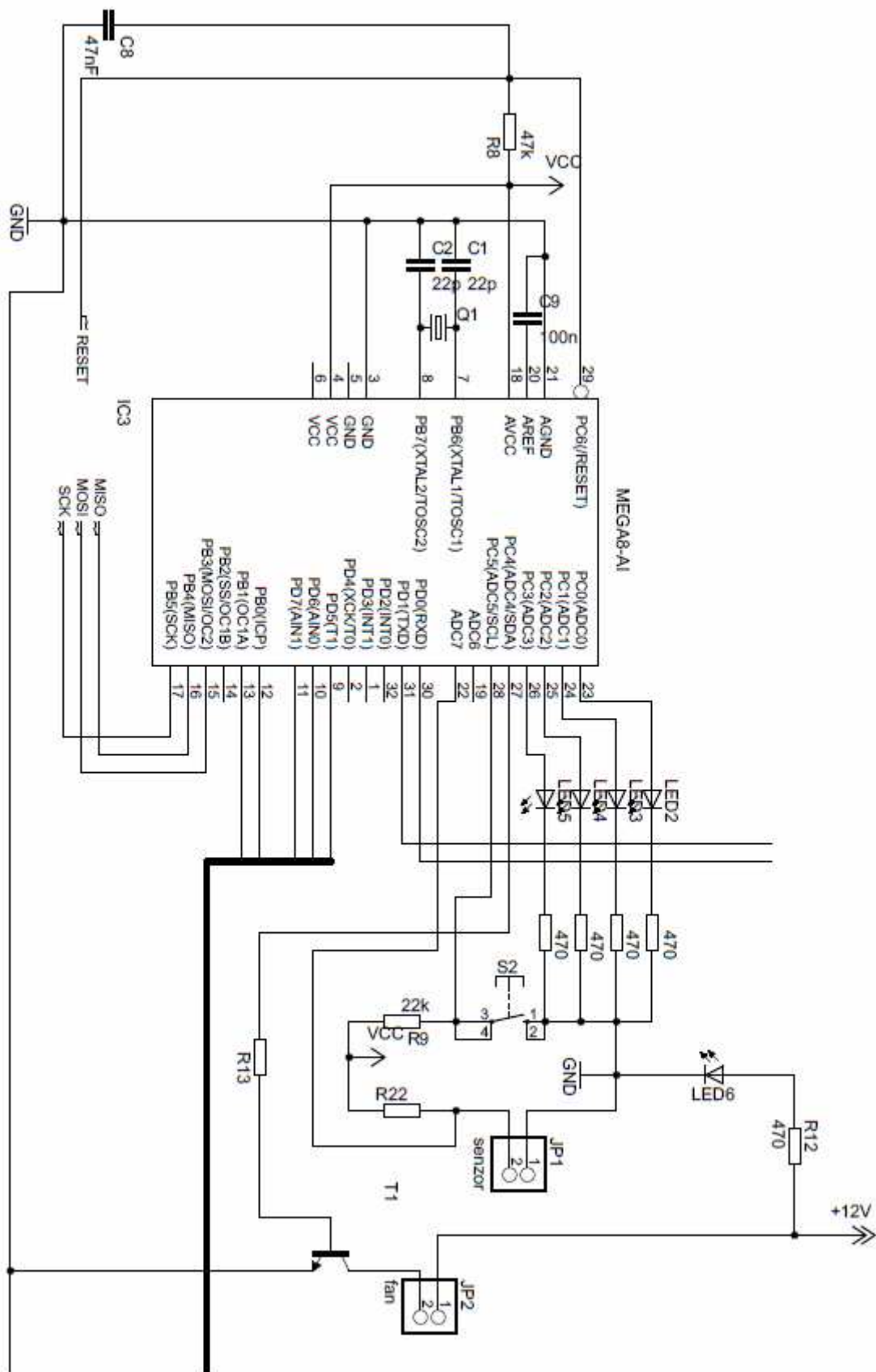
Obrázok č. 27.

Reakcia 78M05 na pripojenie, resp. odpojenie záťaže [6]

Podľa katalógového listu [6] kondenzátor pripojený na výstup zlepšuje prechodovú odozvu, pričom kapacita menšia ako 100nF môže spôsobiť nestabilitu stabilizátora.

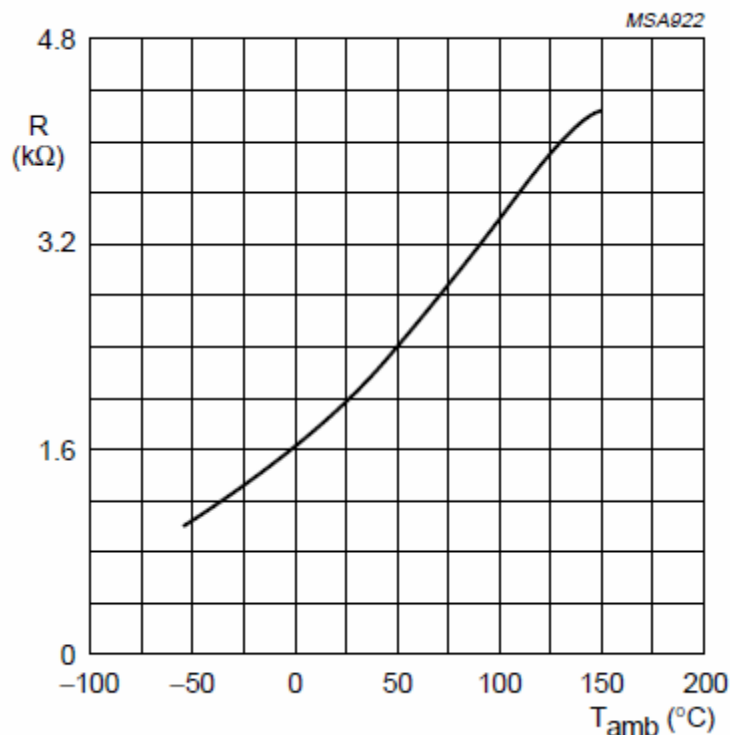
### 3.5 Riadenie

Riadiaca elektronika zabezpečuje vykonávanie samotného riadiaceho algoritmu, ale tiež komunikáciu s nadriadeným počítačom, následnú realizáciu príkazov a signalizáciu stavu elektroniky. Schéma zapojenia je na obrázku č. 28. Riadenie je založené na mikrokontroléry ATmega8 katalógový list [7]. Jeho pracovná frekvencia je 16Mhz. Ako zdroj taktovacieho signálu slúži interný oscilátor riadený externým krištálom Q1. Ako zdroj referenčného napätia je využitý napájací zdroj 5V, ktorý podľa katalógového listu [7] vyžaduje externý kondenzátor. Na porte C, bity 0 až 3, sú pripojené led diódy, ktoré sú využívané na signalizáciu stavu zariadenia. Riadiace piny PB0, PD7, PD6, PD5 sú privedené do L6506 na piny IN1 až IN4, pin PB1 ovláda vstup Power enable. V pamäti mikrokontroléra sú dva programy boot loader a riadiaci program. Boot loader je určený k "napaľovaniu" nového riadiaceho programu. Jeho spustenie je zabezpečované tlačítkom S2. Boot loader sa spustí ak po reštarte mikrokontroléra je tlačítko stlačené. Potom očakáva ďalšie príkazy, ktoré čaká na rozhraní USART. Ale boot loader nedokáže nahradiť programovanie pomocou externých programátorov a preto sú vyvedené aj signály SPI rozhrania.



Obrázok č. 28.  
Schéma riadiacej elektroniky

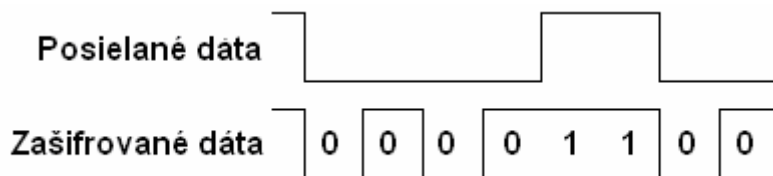
Konektor JP1 slúži na pripojenie snímača teploty pripevneného na chladič spínacích tranzistorov. Ako snímač je použitý KTY81-210. Snímače KTY81-2xx sú kremíkové tepelne závislé snímače s kladným tepelným koeficientom. Odpor pri 25°C je 2000  $\Omega$  s toleranciou 0,5%. Pre presné meranie teploty je potrebné zabezpečiť prúdový zdroj pre napájanie snímača. Všetky údaje v katalógovom liste sú merané pri 1mA. Ale pre meranie teploty chladiča a spínanie ventilátora nie je potrebná veľká presnosť. Odpor R22 je navrhnutý tak, aby prúd pretekajúci snímačom bol pri pokojovej teplote, približne 1mA. JP2 je konektor pre ventilátor pripevnený na chladič. Ventilátor je spínaný tranzistorom zapojeným ako prúdový zosilňovač.



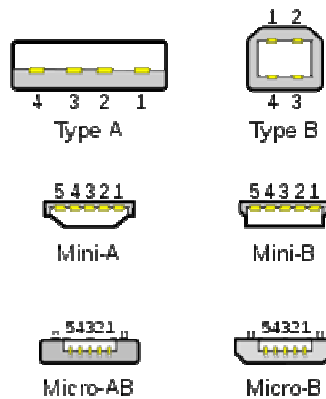
Obrázok č. 39  
Závislosť odporu snímača na jeho teplote  
pri konštantnom prúde I=1mA [10]

### 3.6 USB

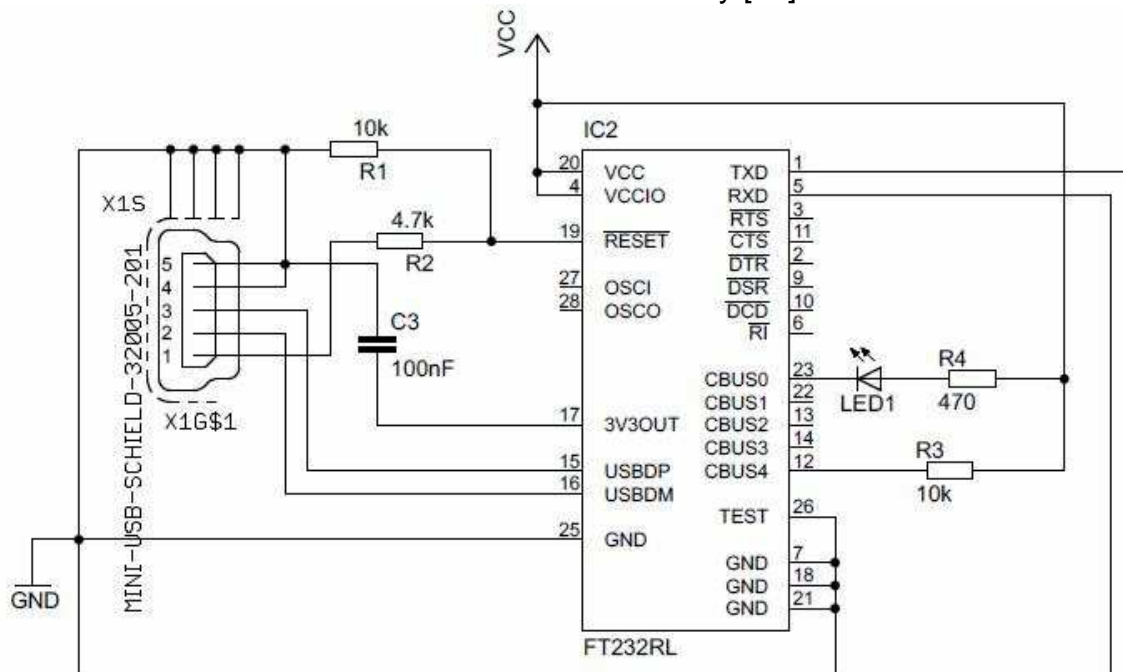
Zariadenie komunikuje s nadriadeným počítačom pomocou zbernice USB. USB je sériová zbernica, na ktorej je komunikácia riadená masťom z pravidla PC. Na zbernici môže byť pripojených až 127 zariadení. Každé zariadenie dostane vlastnú adresu, teda je možné cielene komunikovať s vybraným zariadením. Komunikáciu vždy zahajuje zariadenie typu master. Dáta sa prenášajú v krátkych 8-bajtových alebo dlhých 256-bajtových paketoch. Každý z dátových paketov má presne vyhradenú 1ms na prenos po zbernici. Synchronizácia hodinových signálov je zabezpečovaná šifrovaním prenášaných dát pomocou NRZI - metódy. Pri tomto type šifrovania sa mení úroveň signálu ak sa posielá log 0. Log 1 nevyvoláva žiadnu zmenu. Každý dátový paket začína synchronizačným bajtom 0000 0001b, takže príjemca dostane osem-krát zmenu úrovne a môže sa synchronizovať. [11]



Obrázok č. 30.  
NRZI šifrovanie



Obrázok č. 31.  
Používané USB konektory [12]



Obrázok č. 32.  
Schéma prevodníku USB - UART

Na strane zariadenia je komunikácia zabezpečená prevodníkom FT232RL. Tento s mikrokontrolérom komunikuje pomocou protokolu UART a dáta prenáša na USB zbernicu. Podľa katalógového listu [8] je prevodník možné napájať buď z USB zbernice, alebo z cieľovej aplikácie. V prípade napájania z aplikácie je treba po pripojení k USB FT232RL resetovať. Toto je zabezpečené odporovým deličom R1 a R2. Použitie zapojenie je doporučené výrobcom pre tento spôsob napájania [8]

obrázok č. 32. Piny CBUS 0 až 4 sú programovateľné. Môžu slúžiť na signalizáciu stavu zbernice, alebo ako riadiace vstupy, výstupy tabuľka č. 17. Na strane počítača sú dve možnosti ako posielat' dáta. Prvým spôsobom je komunikácia priamo s prevodníkom pomocou USB rozhrania. Druhým spôsobom je vytvorenie virtuálneho COM portu. Potom z pohľadu aplikácie komunikácia prebieha na štandardnom UART rozhraní.

Tabuľka č. 17. Programovateľné piny FT232RL [8]

| Názov signálu | Použiteľné piny                   | význam   |
|---------------|-----------------------------------|--|
| TXDEN         | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Povolenie vysielania dát po RS485  |
| PWREN#        | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Tento signál je nastavený na 0 ak je pripojená USB zbernica. Ak je USB zbernica v spánku hodnota je log 1. Využíva sa na spínanie MOSFET tranzistora, ako spínač napájania externej logiky. Vyžaduje externý Pull-up rezistor. |
| TXLED#        | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Signalizuje vysielanie dát cez USB. Počas vysielania dát je nastavený na log 0.  |
| RXLED#        | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Signalizuje prijímanie dát cez USB. Počas prijímania dát je nastavený na log 0.  |
| LX&RXLED#     | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Signalizuje komunikáciu po USB. Počas prenosu dát je nastavený na log 0.   |
| SLEEP#        | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Počas režimu spánku je nastavený na log 0. Zvyčajne slúži k vypnutiu externej logiky prevodníka úrovni RS232   |
| CLK48         | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Výstup hodinového signálu 48Mhz  |
| CLK24         | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Výstup hodinového signálu 24Mhz  |
| CLK12         | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Výstup hodinového signálu 12Mhz  |
| CLK6          | CBUS0, CBUS1, CBUS2, CBUS3, CBUS4 | Výstup hodinového signálu 6Mhz   |
| CBitBangI/O   | CBUS0, CBUS1, CBUS2, CBUS3        | Signál zbernice v režime BitBang   |
| BitBangWRn    | CBUS0, CBUS1, CBUS2, CBUS3        | Signál zbernice v režime BitBang   |
| BitBangRDn    | CBUS0, CBUS1, CBUS2, CBUS3        | Signál zbernice v režime BitBang   |

## 4.0 Riadiaci software

Riadiaci software je napísaný v jazyku C. Ako GCC kompilátor je použitý WinAVR. Ako vývojové prostredie je použité integrované vývojové prostredie AVR-Studio. Oba dva tieto produkty sú voľne dostupné. AVR-studio poskytuje výrobca použitého mikrokontroléra. WinAVR je projekt GNU.

### 4.1 Riadiaci program v mikrokontroléry

Riadiaci program využíva štyri prerušenia. Používa všetky tri časovače a štvrtým prerušením je príjem dát z PC. Okrem týchto prerušení má ešte tri funkcie posielanie, `init_mcu` a `main`. Základná funkcia `main` vykonáva tri úlohy.

1. Pri spúšťaní zavolá funkciu `init_mcu()`, čím sa nastaví potrebné registre pre správnu činnosť hardveru.
2. Zisťuje, či motor dosiahol požadovanú rýchlosť. Ak nie, svieti Led 3 a tým signalizuje, že rýchlosť motora sa mení.
3. Meria teplotu chladiča spínacích tranzistorov ako úbytok napätia na snímači. Napätie je merané na kanále ADC7. Ako referenčný zdroj sa využíva napájacie napätie AD prevodníka, 5V. Nastavená citlivosť AD prevodníka 1024 bitov. Takže vzťah na výpočet hodnoty registra ADCW v závislosti na odpore snímača je:

$$ADCW = \frac{V_{cc}(R_{22} + R_s)}{R_s} \frac{1024}{V_{cc}} = \frac{1024(3300 + R_s)}{R_s}$$

Hodnoty odporu  $R_s$  sa dajú získať z katalógového listu KTY81-210 [10]. Funkcia na základe nameranej teploty ovláda ventilátor. Ventilátor sa zapne ak hodnota registra je vyššia ako 414, čo zodpovedá približne 40°C. K vypnutiu ventilátora príde, ak je nameraná hodnota menšia ako 395, čo zodpovedá približne 30°C.

Funkcia posielanie (`unsigned char data`) má za úlohu poselať znak data riadiacemu počítaču.

Časovač 0 je nastavený na 1 ms a využíva sa na zabezpečenie požadovaného sklonu nábežnej a dobežnej rampy. Zadávaným parametrom je `int delta`. Hodnota `delta` vyjadruje o koľko sa zmení perióda krokovania za 1ms. Najmenšia nastaviteľná zmena je 10  $\mu$ s, ak je rozdiel medzi žiadanou a okamžitou rýchlosťou menší ako hodnota `delta`, okamžitá rýchlosť sa nastaví na požadovanú rýchlosť. Okamžitá rýchlosť sa nastavuje na žiadanú aj v prípade ak je hodnota `delta` = 0.

Časovač 1 je nastavený na 10  $\mu$ s. Jeho úlohou je generovať riadiaci signál v požadovanej perióde. Vstupné parametre sú `long int cas_realny`, `long int t` a `char smer`. `Cas_realny` vyjadruje periódu kroku a je to počet opakovaní prerušenia časovača 1 bez zmeny riadiacich signálov. Hodnota `t` označuje čas, ktorý uplynul od predchádzajúcej zmeny signálov. Hodnota `smer` označuje zmysel otáčania motora. Maximálna podporovaná hodnota je 1000ms. Generovaný riadiaci signál



je štvortaktný dvoj fázový. Períodu krokovania môžeme vypočítať ako násobok času potrebného na jedno prerušenie a hodnoty nastavenej v premennej cas\_realny.

$$T = \text{cas\_realny} \cdot 0.00001 \text{ [s]}$$

Obsluha prerušenia časovača 1:

```
ISR(TIMER1_OVF_vect)
```

```
{
```

```
TCNT1L=0x8D;
```

```
TCNT1H=0xFF;          //10uS + korekcia
```

```
t++;
```

```
if (t>=cas_realny){
```

```
    if (smer==0){
```

```
        if (c==0){
```

```
            clear_bit(PORTB,0);set_bit(PORTD,5); //Takt 1
```

```
        }
```

```
        if (c==1){
```

```
            clear_bit(PORTD,5);set_bit(PORTD,6); //Takt 2
```

```
        }
```

```
        if (c==2){
```

```
            clear_bit(PORTD,6);set_bit(PORTD,7); //Takt 3
```

```
        }
```

```
        if (c==3){
```

```
            clear_bit(PORTD,7);set_bit(PORTB,0); //Takt 4
```

```
        }
```

```
    }
```

```
    else{
```

```
        if (c==0){
```

```
            clear_bit(PORTD,5);set_bit(PORTB,0); //Takt 1
```

```
        }
```

```
        if (c==1){
```

```
            clear_bit(PORTB,0);set_bit(PORTD,7); //Takt 2
```

```
        }
```

```
        if (c==2){
```

```
            clear_bit(PORTD,7);set_bit(PORTD,6); //Takt 3
```

```
        }
```

```
        if (c==3){
```

```
            clear_bit(PORTD,6);set_bit(PORTD,5); //Takt 4
```

```
        }
```

```
    }
```

```
c++;
```

```
if (c==4)
```

```
    c=0;
```

```
t=0;
```

```
}
```

```
}
```

Časovač 2 pravidelne posiela hodnotu okamžitej rýchlosti riadiacemu PC. Pretože okamžitá rýchlosť je premenná typu long int a poselať je možné iba jeden bajt je nutné ho rozložiť. Pre potreby zobrazenia postačuje rozlišovanie 100µs. Čo za predpokladu, že maximálna hodnota je 1s, znamená, že maximálna hodnota posielaných údajov je 10000. Na poslanie takýchto malých čísel stačia 2 bajty. Rozlišovanie smeru je zabezpečené rozdelením intervalu 20000 na kladný a záporný s nulou v 10000. Nula je signalizovaná špeciálnou hodnotou 20001.

Obsluha prerušenia časovača 2:

```
ISR(TIMER2_OVF_vect)
{
    int data=0;
    unsigned char* p_data;    // definovanie smerníka typu unsigned char

    tx_poc++;
    if (tx_poc>=30)
    {
        if (cas_realny!=0)
        {
            data=(cas_realny/10);
            if (smer==0)
                data=data+10000;
        }
        else
            data=20001;        // cas_realny = 0

        p_data=(unsigned char*)&data; //nastavenie adresy smerníka

        posielanie(*p_data);        // posielanie prvého bajtu
        posielanie(*(p_data+1));    // posielanie druhého bajtu
        tx_poc=0;
    }
}
```

Príjem riadiacich príkazov je zabezpečený obsluhou prerušenia príjmu. Podporované príkazy sú nastavenie sklonu rampy, nastavenie rýchlosti, pripojenie motora a identifikácia elektroniky.

Príkaz na nastavenie sklonu rampy je D. Po prijíme tohto príkazu program očakáva 2-bajtovú hodnotu sklonu rampy. Nová hodnota sa ukladá postupne do premennej x\_delta a po skončení prenosu sa skopíruje do premennej delta, vynuluje sa počítadlo a program je pripravený na príjem ďalšieho príkazu.

Obsluha príkazu D

```
if (prikaz=='D')
{
    p_XX=(unsigned char*)&X_delta;
    *(p_XX+index_primu)=k;
    index_primu++;
}
```

```

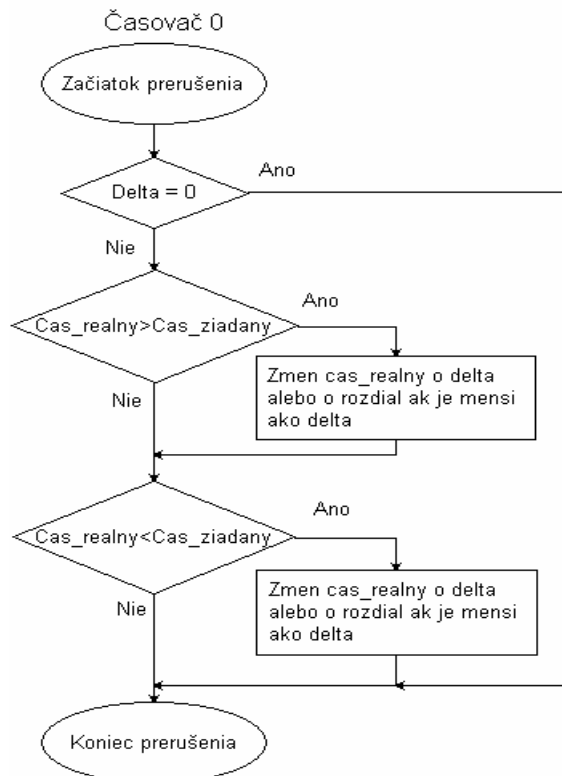
if (index_primu==2)
{
    delta=X_delta;
    prikaz=0;
    index_primu=0;
}
}

```

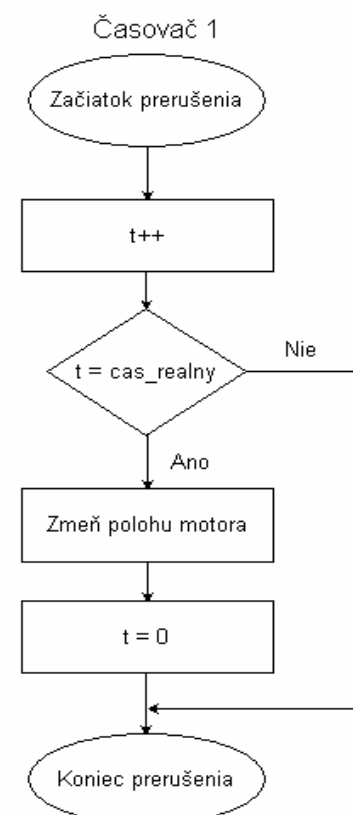
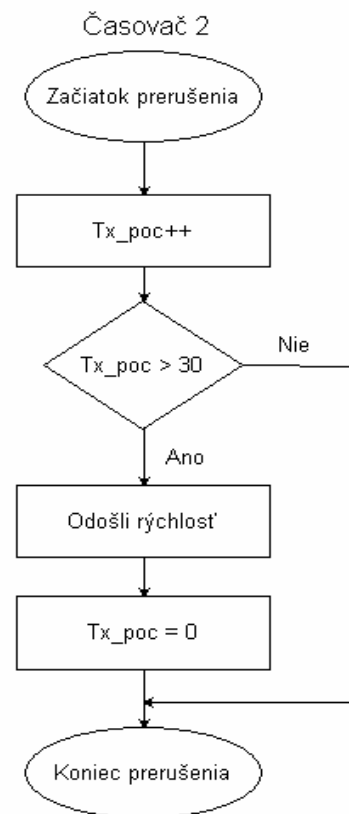
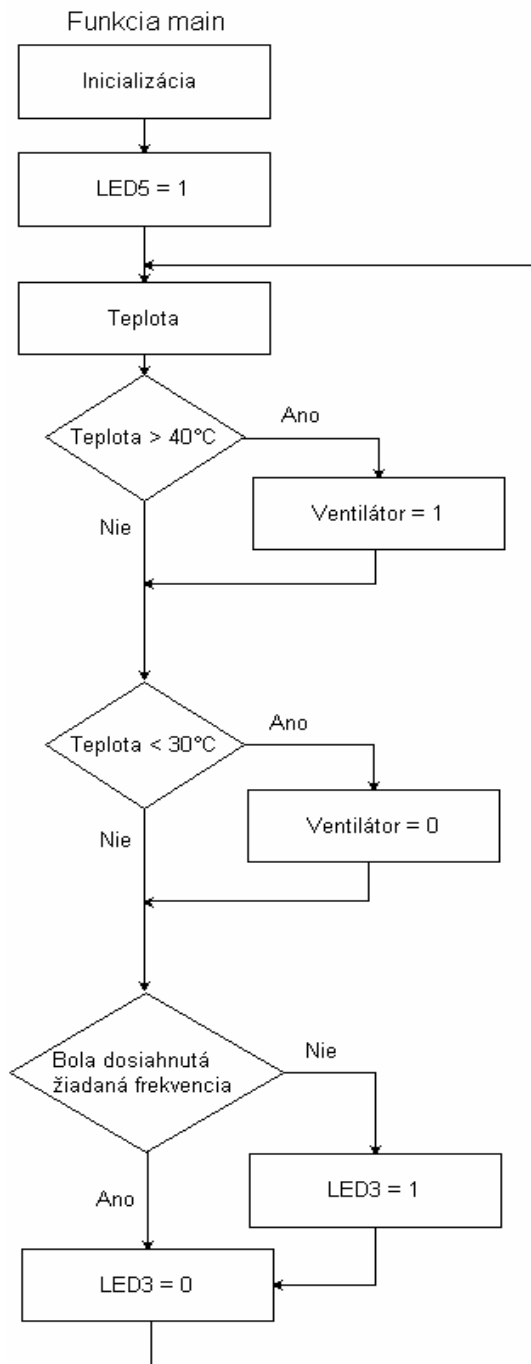
Príkaz pre nastavenie rýchlosti je S. Po prijíme tohto znaku program očakáva príjem 4-bajtovej hodnoty, ktorá označuje smer a novú rýchlosť. Hodnota sa prenáša podobne ako u príkazu D. Smer je identifikovaný znamienkom prijatých dát. Ak je znamienko kladné smer je 0, ak je záporné smer je 1. Ak sa mení smer otáčania motora, je potrebné zmeniť počítadlo taktu na rovnaké nastavenie v reverznom režime. Počítadlo taktu je v premennej c. Napríklad ak sa mení smer z 0 na 1 a aktuálna hodnota c je 1 tak je treba c nastaviť na hodnotu 3. Ak je požadovaná rýchlosť nula obsluha príkazu vypne časovač 0 a 1. Premennú t nastaví na nulu. Ak je požadovaná zmena rýchlosti a motor nie je pripojený, ignoruje sa nastavenie rampy a priamo sa nastaví perióda riadiaceho signálu.

Príkaz pripojenia motora je D. Tento príkaz nemá žiadne ďalšie parametre a po jeho prijíme mikrokontrolér neguje signál PE, čím povolí respektíve zakáže napájanie motora. Ako signalizácia tohto stavu slúži Led 2. Ak Led 2 svieti motor je pripojený.

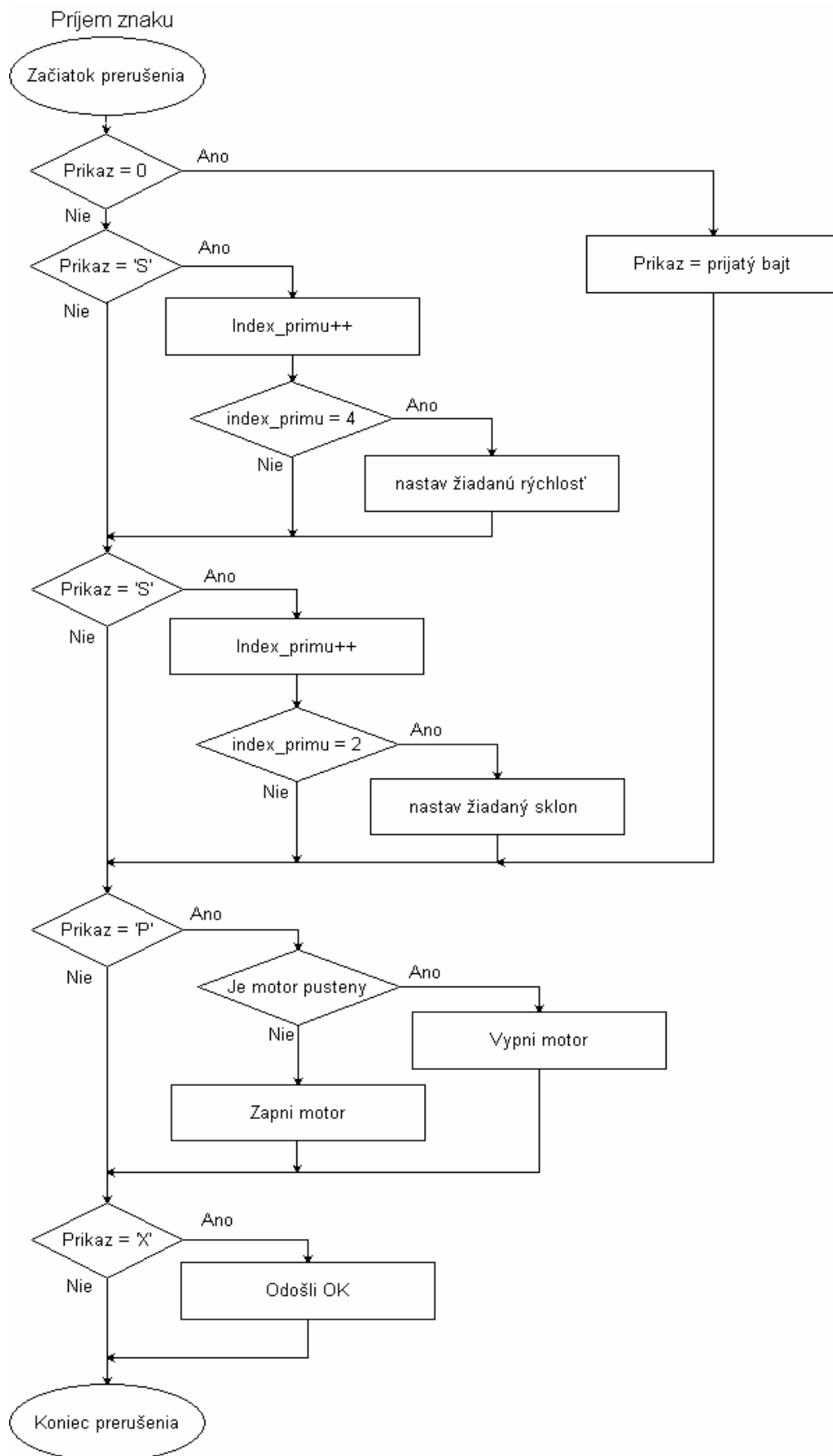
Príkazom X sa zariadenie identifikuje. Po príchode tohto znaku mikrokontrolér odpovie odoslaním dvoch znakov OK.



Vývojový diagram 1  
Obsluha prerušenia časovača 0



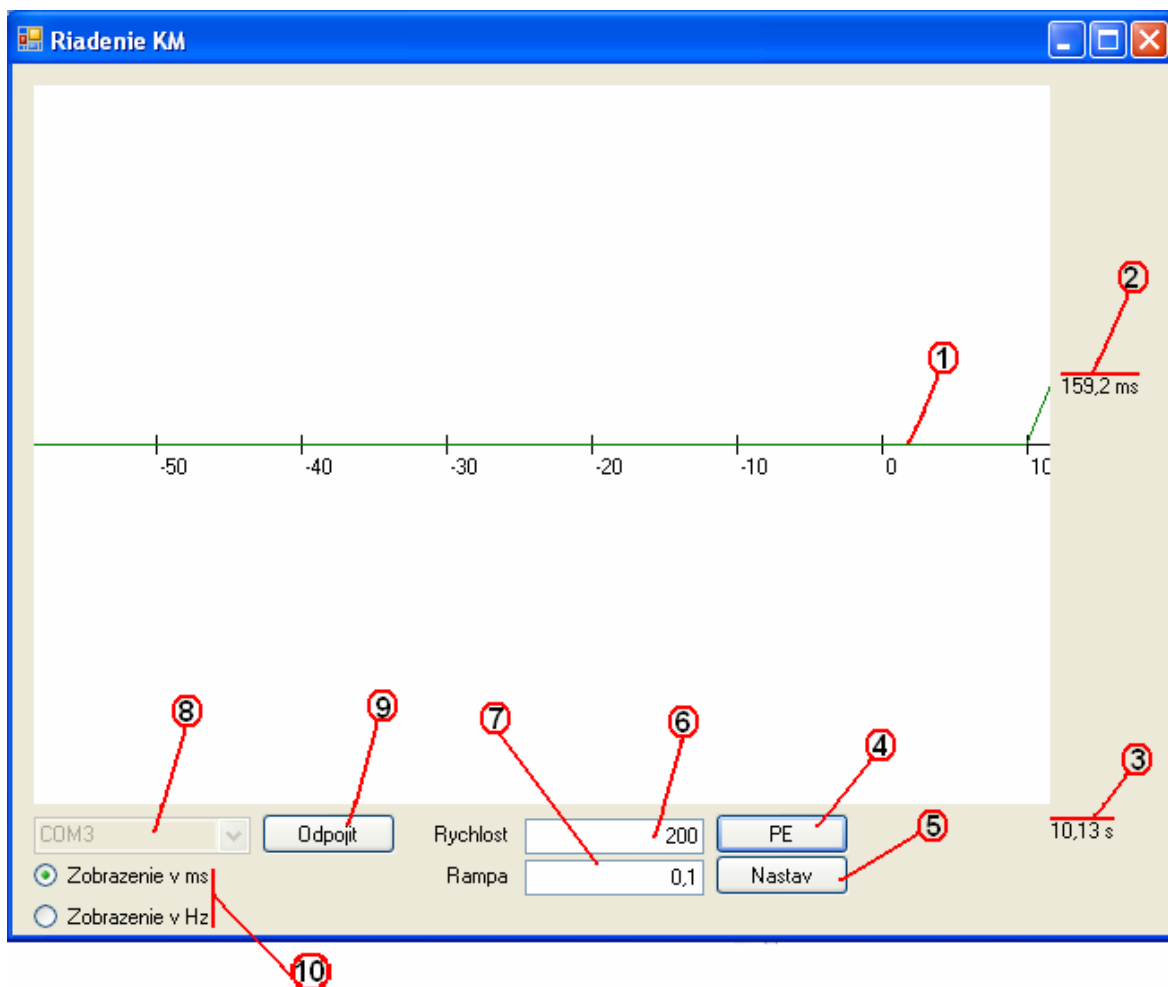
Vývojové diagramy 2  
Funkcia main prerušenie časovača 1 a časovača 2



Vývojový diagram 3  
Prerušenie príjmu znaku.

## 4.2 Ovládací program v PC

Ovládací program komunikuje s užívateľom a odovzdáva riadiacej elektronike jeho požiadavky. Program preloží požiadavky užívateľa do príkazov popísaných v predchádzajúcej časti a odošle ich riadiacej elektronike. Užívateľ dostáva spätnú väzbu v podobe meniaceho sa priebehu rýchlosti motora. Graf zobrazujúci tento priebeh môže byť vykresľovaný v dvoch režimoch.



Obrázok č. 33.  
Ovládací program

- 1) Časová os
- 2) Os rýchlosti motora
- 3) Poloha kurzora myši na časovej osy
- 4) Tlačidlo PE
- 5) Tlačidlo Nastav
- 6) Rýchlosť motora (perióda riadiaceho signálu v ms)
- 7) Sklon rampy (o koľko ms sa ma perióda riadiaceho signálu zmeniť za 1 ms)
- 8) Komunikačný port
- 9) Tlačidlo Pripojiť / Odpojiť
- 10) Nastavenie jednotky osi rýchlosti motora

### 4.3 Návod

**Pripojenie k riadiacej elektronike:** Po spustení ovládacieho programu je nutné zvoliť komunikačný port. Všetky dostupné porty sú načítané v ComboBoxe na obrázku č. 33 položka 8. Po zvolení portu klikneme na tlačidlo Pripojiť položka 9. Ak sa podarilo pripojiť k riadiacej elektronike text na tlačidle sa zmení na Odpojiť, sprístupnia sa tlačidlá PE a Nastav, a Nastavený port v ComboBoxe sa nedá zmeniť. Pre odpojenie stačí stlačiť tlačidlo Odpojiť položka 9.

**Nastavenie počiatočnej rýchlosti:** Rýchlosť sa nastavuje v TextBoxe Rýchlosť položka 6. Zadávaná hodnota je perióda riadiaceho signálu v ms a program akceptuje dve desatinné miesta napríklad 2,56. Rozsah povolených hodnôt je od -1000,00 do 1000,00 ms. Po stlačení tlačidla Nastav položka 5, program odošle nastavenú hodnotu riadiacej elektronike.

**Spustenie motora:** Motor sa spustí stlačením tlačidla PE položka 4. Po spustení motora elektronika začne posielať informácie o okamžitej rýchlosti motora a ovládací program ich začne vykresľovať do grafu. Graf zobrazuje priebeh v dvoch režimoch. Prednastavený režim je v milisekundách. Teda y-ová os je v milisekundách. V druhom režime je y-ová os v hercoch. Aktuálna rýchlosť motora sa zobrazuje v nastavených jednotkách položka 2. Poloha položky sa tiež mení v závislosti na aktuálnej rýchlosti. Zobrazované jednotky sa môžu zmeniť výberovými tlačidlami položka 10.

**Zmena rýchlosti motora:** Novú rýchlosť zadáme znova do TextBoxu Rýchlosť a odošleme stlačením tlačidla Nastav. Ak chceme aby sa motor dostal do novej rýchlosti po rampe s určitým sklonom zadáme aj sklon rampy do TextBoxu Rampa položka 7. Zadávaná hodnota reprezentuje o koľko ms sa zmení perióda riadiaceho signálu za 1ms. Rozsah zadávaných hodnôt je od 0 do 655,36 ms. Ak chceme aby si riadiaca elektronika uchovala aktuálne nastavenie vymažeme hodnotu TextBoxu. Pri reverzácii bude riadiaca elektronika sklon rampy ignorovať.

**Držanie polohy:** Pre držanie polohy stačí zadať do TextBoxu Rýchlosť hodnotu 0 v tomto prípade riadiaca elektronika tiež ignoruje sklon rampy a motor okamžite zastaví.

**Odpojenie motora:** Odpojenie motora je možné stlačením tlačidla PE. Po odpojení motora prestane elektronika posielať údaje ovládacímu programu a graf sa zastaví. Po znovu pripojení sa motor rozbehne v režime nastavenom pred odpojením.

**Reverzácia:** Reverzáciu vykoná motor ak sa okamžite (bez straty kroku) zmení zmysel otáčania hriadele. Reverzáciu je možné vykonávať len do určitej frekvencie, táto frekvencia sa nazýva frekvencia reverzácie.

**Rezonančné zóny motora:** Tento jav sa prejavuje stratou synchronizácie motora pri určitej frekvencii riadiaceho signálu. Tuto frekvenciu zistíme postupným zvyšovaním frekvencie napríklad začneme na 1Hz so sklonom rampy 0,01 a necháme frekvenciu rásť do 500Hz.

**Rozbehová frekvencia:** Je maximálna frekvencia, pri ktorej sa motor dokáže roztočiť z kľudového stavu bez straty synchronizácie.

## Záver

Cieľom tejto bakalárskej práce bolo navrhnuť a realizovať akčný člen pre unipolárne riadenie krokových motorov. Riadiaca elektronika je ovládaná nadriadeným PC. PC a elektronika komunikujú obojsmerne po USB zbernici.

Výkonovú časť sme vyrobili pomocou diskretných súčiastok, unipolárnych tranzistorov. Unipolárne tranzistory sme zvolili kvôli lepším tepelným vlastnostiam a kvôli menším nárokom na budenie výkonovej časti. Ako prúdová ochrana bol použitý integrovaný obvod. Integrovaný kontrolór prúdu zabezpečuje odpojenie preťaženej fázy motora. Prúdový chránič obmedzuje aj špičky spôsobené skokovým pripájaním vinutia motora. Maximálna veľkosť prúdu je nastaviteľná podľa požiadaviek konkrétneho krokového motora. Chladenie výkonovej časti je aktívne a je riadené na základe teploty pasívneho chladiča spínacích tranzistorov. Rozsah teplôt je programovateľný.

Riadenie výkonových častí je realizované 8-bitovým mikrokontrolérom architektúry AVR. Konkrétne typom ATmega8. Má dostatok vývodov na realizáciu riadenia aj signalizáciu stavu elektroniky. Jeho pamäť je dosť veľká pre program aplikácie a zavádzača, ktorý uľahčuje programovanie mikrokontroléra. Pre pripojenie k zbernici USB sme použili prevodník USB – UART, po pripojení riadiacej elektroniky sa na nadriadenom PC vytvorí virtuálny COM port.

Riadiaci program v PC má za úlohu odovzdávať riadiacej elektronike prostredníctvom príkazov požiadavky užívateľa. Nevhodné, alebo nerealizovateľné požiadavky odmieta a vyzýva k oprave zadávaných hodnôt. Ďalšou úlohou riadiaceho programu je zobrazovať priebeh rýchlosti otáčania, alebo periódy riadiaceho signálu. Súčasťou riadiaceho programu je aj návod, ktorý popisuje základnú prácu s programom, ako aj postupy na získavanie niektorých vlastností pripojeného motora.

Výsledný produkt tejto bakalárskej práce je vyrobený a otestovaný riadiaci systém, ktorý ovláda unipolárne dvojfázové krokové motory. Používaný algoritmus je štvortaktný a v jednom takte sú pripojené dve fázy. Riadiaca elektronika je umiestnená v plastovej krabičke.



## Zoznam použitej literatúry

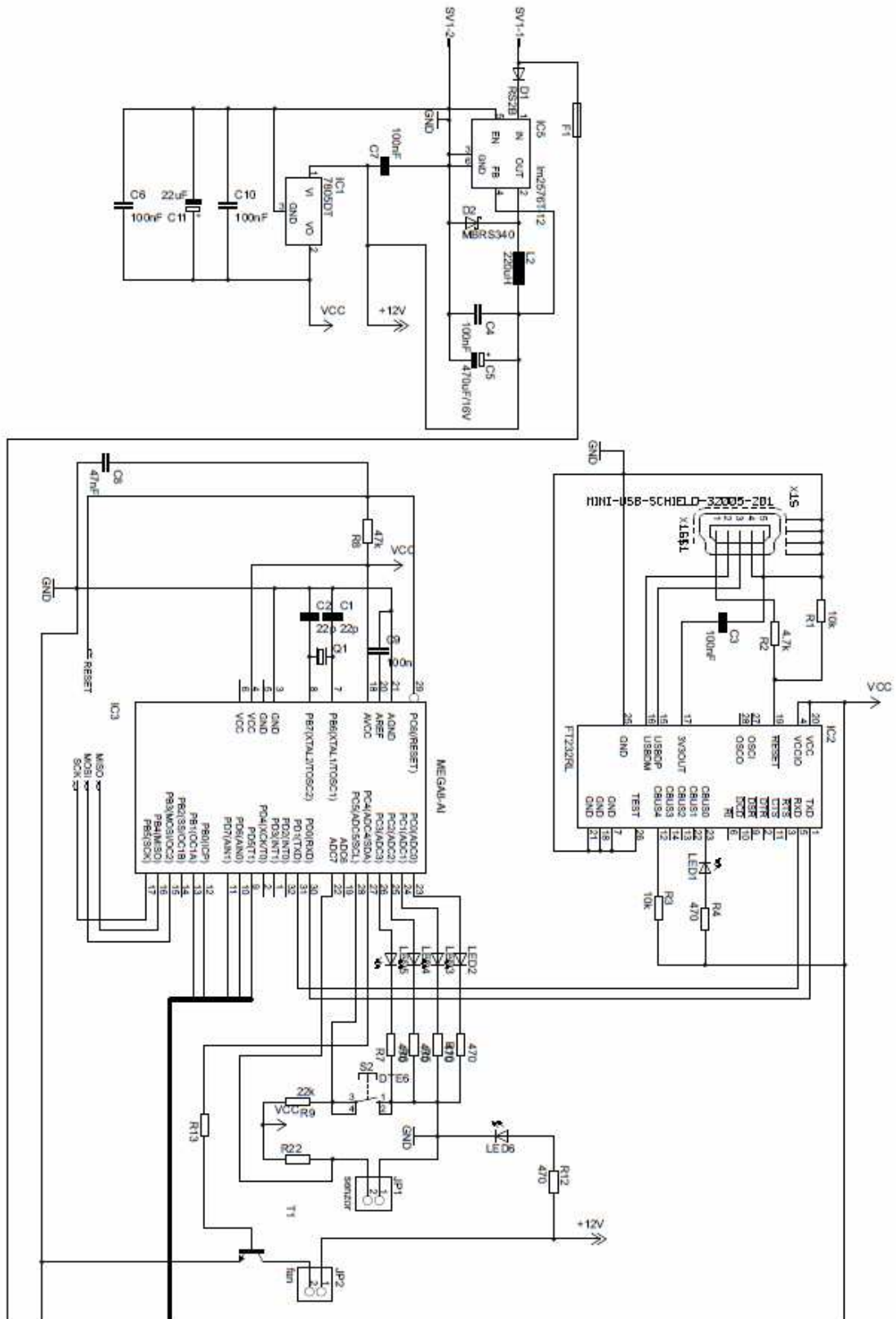
- [1] International Rectifier, 2003, irlr2908 datasheet,  
<http://www.irf.com/product-info/datasheets/data/irlr2908.pdf>
- [2] Microchip Technology Inc., 2006, TC4427 datasheet,  
<http://ww1.microchip.com/downloads/en/DeviceDoc/21422d.pdf>
- [3] STMicroelectronics, 2003, L6506 datasheet,  
<http://www.st.com/stonline/products/literature/ds/1392.pdf>
- [4] National Semiconductor Corporation, 2004, LM2576 datasheet  
<http://www.national.com/ds/LM/LM2576.pdf>
- [5] Fairchild Semiconductor Inc., 2001, MBRS340 datasheet  
<http://www.fairchildsemi.com/ds/MB%2FMBRS340.pdf>
- [6] STMicroelectronics, 2009, 78Mxx datasheet,  
<http://www.st.com/stonline/products/literature/ds/2147/l78m05ab.pdf>
- [7] Atmel Corporation, 2009, ATmega8 datasheet  
[http://www.atmel.com/dyn/resources/prod\\_documents/doc8159.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc8159.pdf)
- [8] Future Technology Devices International Ltd., 2009, FT232RL datasheet  
[http://www.ftdichip.com/Documents/DataSheets/DS\\_FT232R\\_V205.pdf](http://www.ftdichip.com/Documents/DataSheets/DS_FT232R_V205.pdf)
- [9] Vít Olmr, 2005, HW server představuje - Sériová linka RS-232 [online],  
[cit. 2.12.2009], <http://hw.cz/rs-232>
- [10] NXP Semiconductors Netherlands B.V., 2008, KTY81 series datasheet  
[http://www.nxp.com/documents/data\\_sheet/KTY81\\_SER.pdf](http://www.nxp.com/documents/data_sheet/KTY81_SER.pdf)
- [11] Burkhard Kainka, 2002, USB měření, řízení a regulace pomocí sběrnice USB,  
1. vyd. Praha, BEN-Technická literatura, 2002, ISBN 80-7300-073-3
- [12] gadgetophilia, USB 1.0, USB 2.0, USB 3.0 : Guide [online], [cit. 2.5.2010],  
<http://gadgetophilia.com/usb-10-usb-20-usb-30-guide>
- [13] Ing. Pavel Rydlo, 2000, Krokové motory a jejich řízení: Studijní texty, Liberec  
Technická univerzita Liberci, 2000, 15s. <http://www.mti.tul.cz/files/ats/krok2.pdf>

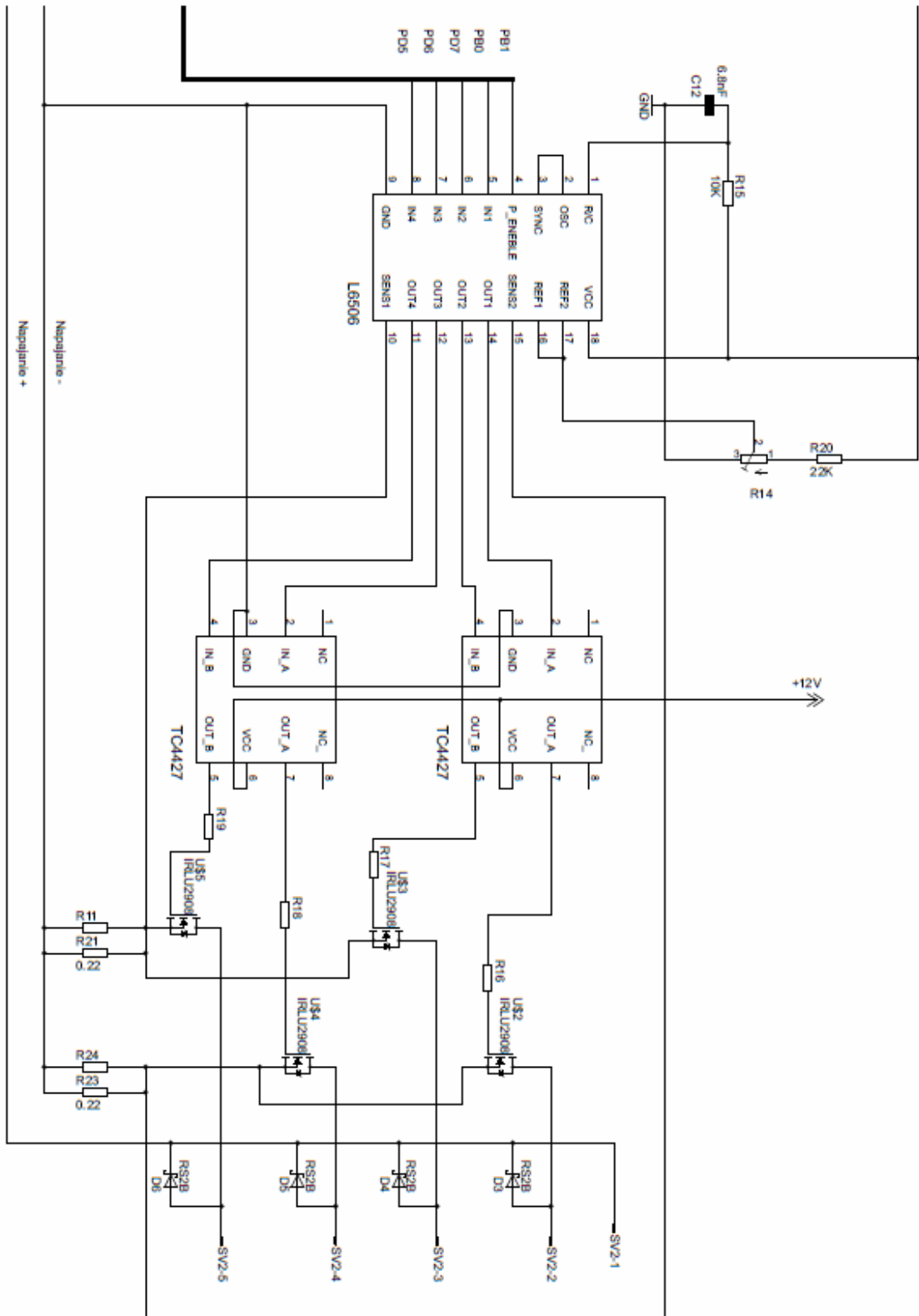
## **Zoznam príloh**

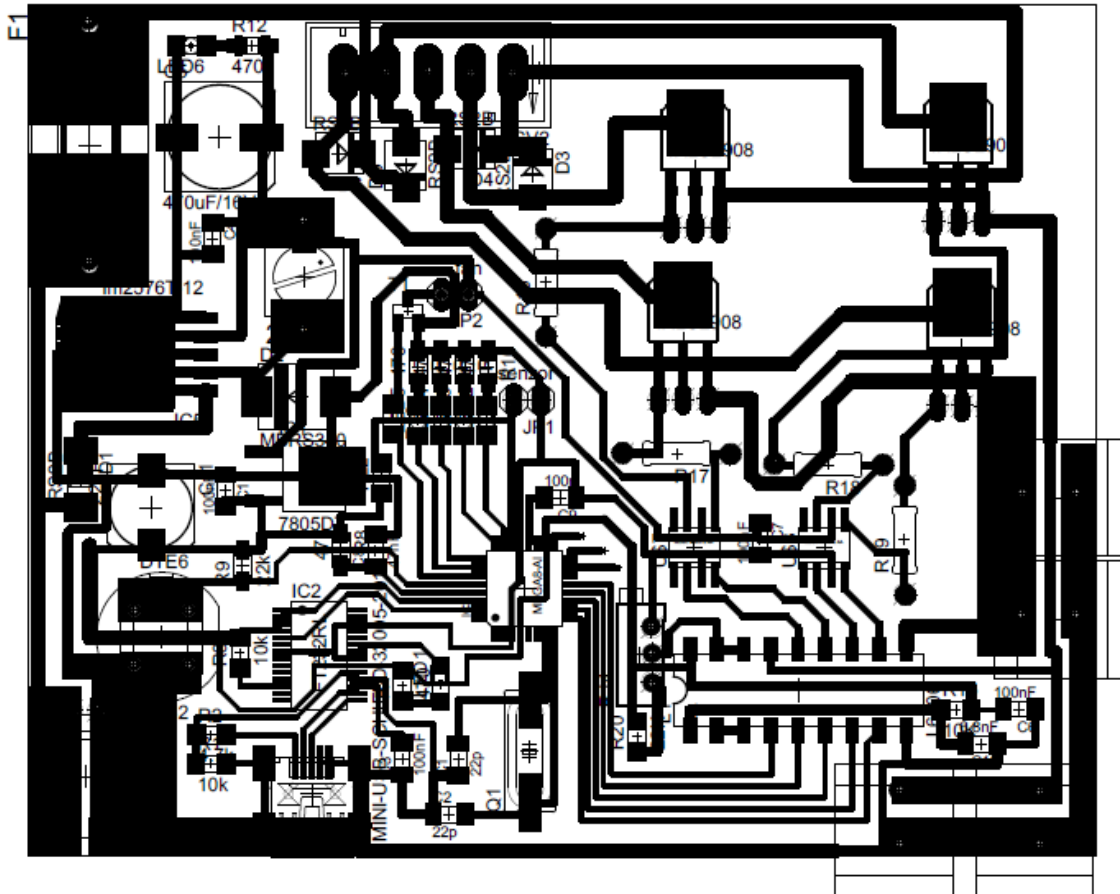
Príloha 1: Schéma a predloha plošného spoja riadiacej elektroniky.

Príloha 2: Obsah priložených CD

Príloha 1:







- Príloha 2:
- Dokument Bakalárska práca
  - Dokument Schéma riadiacej elektroniky
  - Dokument Plošný spoj
  - Zdrojový kód riadiaceho programu pre ATmega8
  - Zdrojový kód ovládacieho programu