Kapitola 1 Úvod 3
1.1 Mikromanipulátory a jejich řešení pomocí piezoelektrického pohonu 3
Kapitola 2 Technický popis systému a problému 5
2.1 Popis řešení generování požadovaného signálu 5
2.2 Popis prototypu zapojení 6
2.2.1 Verze 1 6
2.2.2 Verze 2 8
2.3 Použité součástky 10
2.3.1 Mikroprocesor ATmega32 10
2.3.2 Převodník MAX232 10
2.3.3 Digitální potenciometr AD5235 10
2.3.4 Tranzistory STX 11
Kapitola 3 Program pro mikroprocesor 12
3.1 Nástroje pro programování mikroprocesoru 12
3.2 Program pro zapojení dle verze 1 13
3.2.1 UART rozhraní 13
3.2.2 Obsluha přerušení ukončení příjmu dat 13
3.2.3 SPI rozhraní 14
3.2.4 Procedura pro výpočet intervalu nabíjení 14
3.3 Program pro zapojení dle verze 2 15
3.3.1 Příkazy a časovače 15
3.3.2 Komunikace ze strany PC 16
Kapitola 4 Testování 17
Kapitola 5 Závěr 19
5.1 Dosažené výsledky 19
5.2 Pokračování projektu 20
5.3 Osobní zhodnocení 20
5.4 Použitá literatura a CD přílohy 21

Anotace:

Ve své práci se zabývám návrhem řídícího generátoru pro mikromanipulátor. Princip posunu je typu stick-slip. Jde o generování asymetrického pilovitého využívajíce kapacitního charakteru piezokrystalu.

Nejdříve je uveden teoretický rozbor úlohy, který nám ukáže požadavky na jednotlivé části generátoru.

Dále je uvedeno jakým způsobem byl problém řešen a jsou zde popsány jednotlivé verze návrhu, schémata a fotografie prototypu.

Následuje popis použitých součástek a jejich technické parametry.

Další kapitola popisuje práci s mikroprocesorem. Použité prostředky k jeho programování a popsané důležité úseky kódu.

V kapitole testování jsou zaznamenány výsledky testování zapojení a jeho zhodnocení.

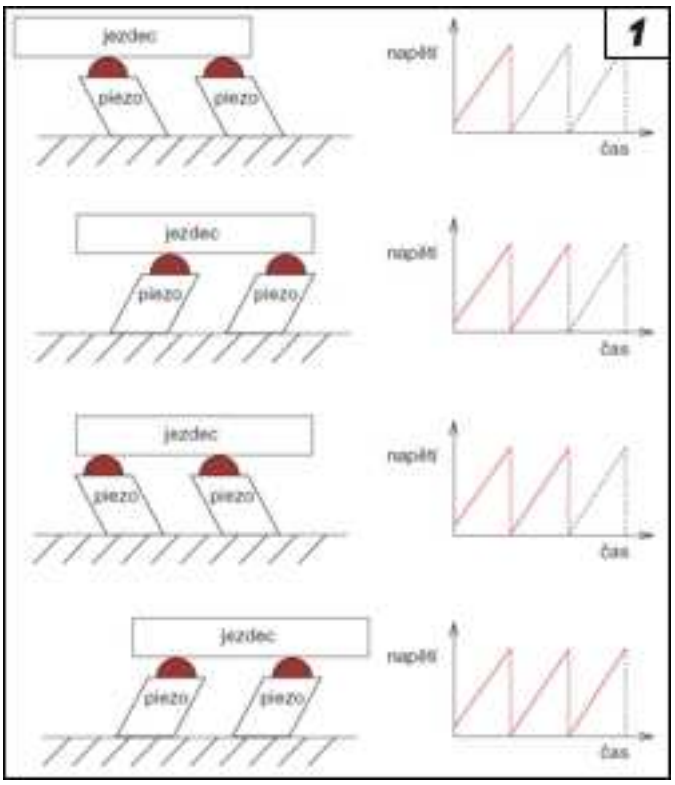
V závěru jsou shrnuty dosažené výsledky a popsány úkoly, které budou následovat.

Kapitola 1 Úvod

1.1 Mikromanipulátory a jejich řešení pomocí piezoelektrického pohonu

Piezoelektrický pohon je znám již z konce 19. století. Použití pro mikromanipulátory se v dnešní době rozšiřuje díky vývoji nových materiálů schopných velkého rozsahu pohybů při velmi malém rozlišení (v řádu desítek nanometrů). Dále je také důležitá linearita vztahu mezi napětím přivedeným na piezočlen a jeho deformací. V této práci se zabývám výrobou prototypu řídícího generátoru pro piezomotor v mikromanipulátoru.

Tento piezomotor je na principu stisk-slip. Jednoduché znázornění tohoto principu lze nalézt na obrázku č.1-1



Obrázek 1-1 Princip stick-slip

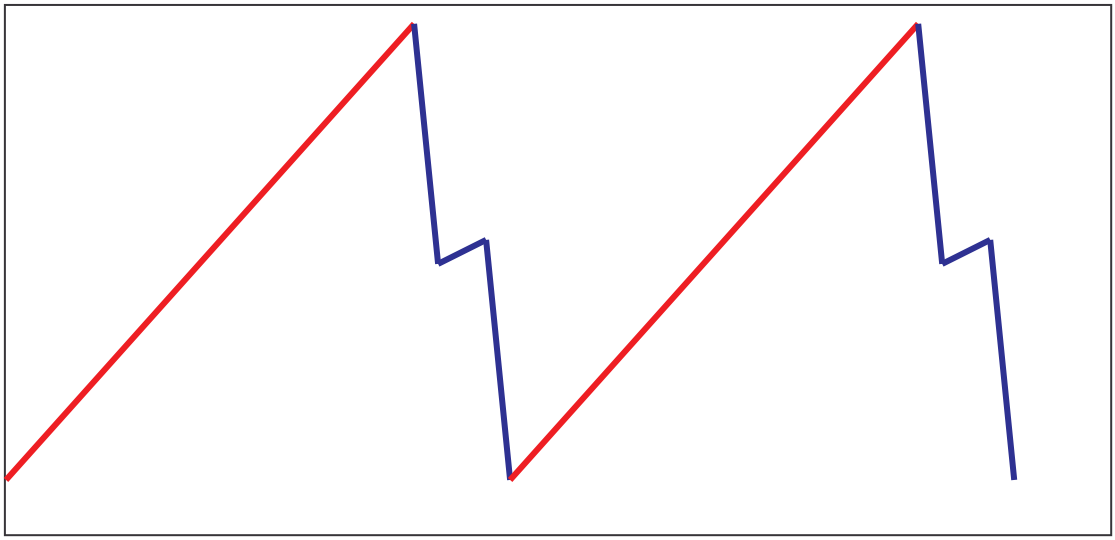
Piezočlen se chová podobným způsobem jako kapacitor, čehož je v našem návrhu využito pro testování, protože v době dokončování práce nebyl prototyp mikromanipulátoru funkční.

Základní myšlenkou je generování napětí přímo na piezočlenu členu. Nejdříve jsme vytvořili obvod spínání generující jednoduchý pilovitý signál.

Dále po zprovoznění jednoduššího generátoru, který generuje jednoduchý asymetrický signál, se bude pokračovat tvorbou složitější varianty s tvarováním signálu typu posicast, který by měl eliminovat oscilaci po fázi proklouznutí.

V našem případě je požadovaný průběh napětí na piezokrystalu zobrazen na obrázku č.1-2. Červeně jsou zvýrazněny části kde se krystal nabíjí a modré kde se vybíjí.

Přesnější informace o dané problematice lze získat z článku [1], ze kterého jsou čerpány informace pro tento odstavec.

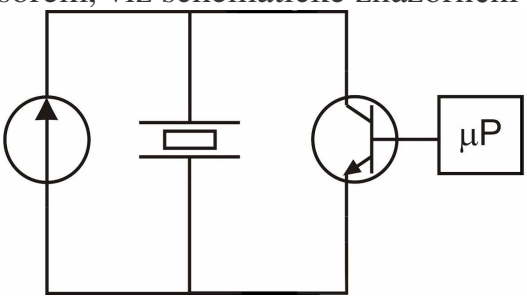


Obrázek 1-2 Ideální průběh napětí

Kapitola 2 Technický popis systému a problému

2.1 Popis řešení generování požadovaného signálu

Napětí na piezočlenu je generováno pomocí dvou řízených zdrojů proudu spínaných mikroprocesorem, viz schématické znázornění na obrázku 2-1.



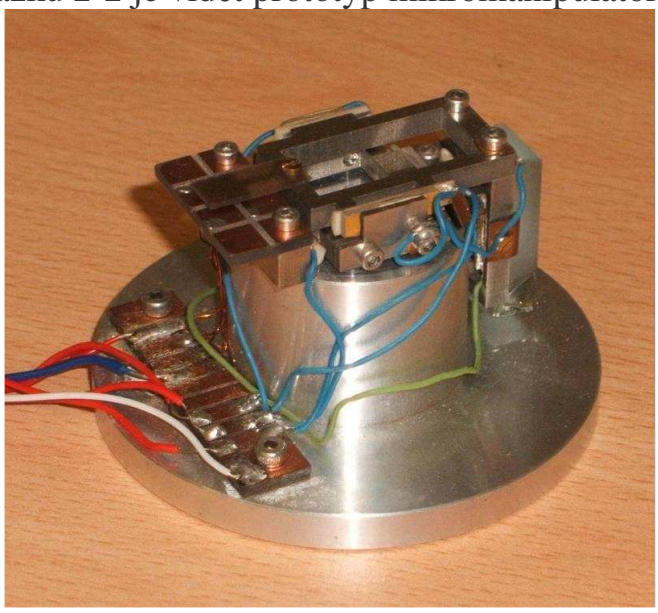
Obrázek 2-1 Schématické znázornění zapojení jednoho zdroje proudu

Pro spínání bylo třeba navrhnout velmi rychlý spínací obvod, aby se docílilo dostatečné strmosti poklesu (růstu) při prokluzu. A dále bylo třeba vybrat dostatečně rychlý mikroprocesor aby byl schopen reagovat ve velmi krátkém čase(jednotky mikrosekund).

Pro řízení byl vybrán mikroprocesor od firmy Atmel s označením ATmega32, umožňující připojení externího krystalu o hodnotě 16MHz, hardwarovým USART a SPI rozhraním, dostatečnými paměťovými rezervami a dostatkem volných I/O pinů.

USART rozhraní je použito pro komunikaci s počítačem a SPI rozhraní pro komunikaci s digitálním potenciometrem řídícím proud dodávaný ze zdroje.

Dále na obrázku 2-2 je vidět prototyp mikromanipulátoru.



Obrázek 2-2 Prototyp mikromanipulátoru

Bohužel tento prototyp zatím není funkční a proto jako jeho náhradu při testování využíváme kondenzátor.

2.2 Popis prototypu zapojení

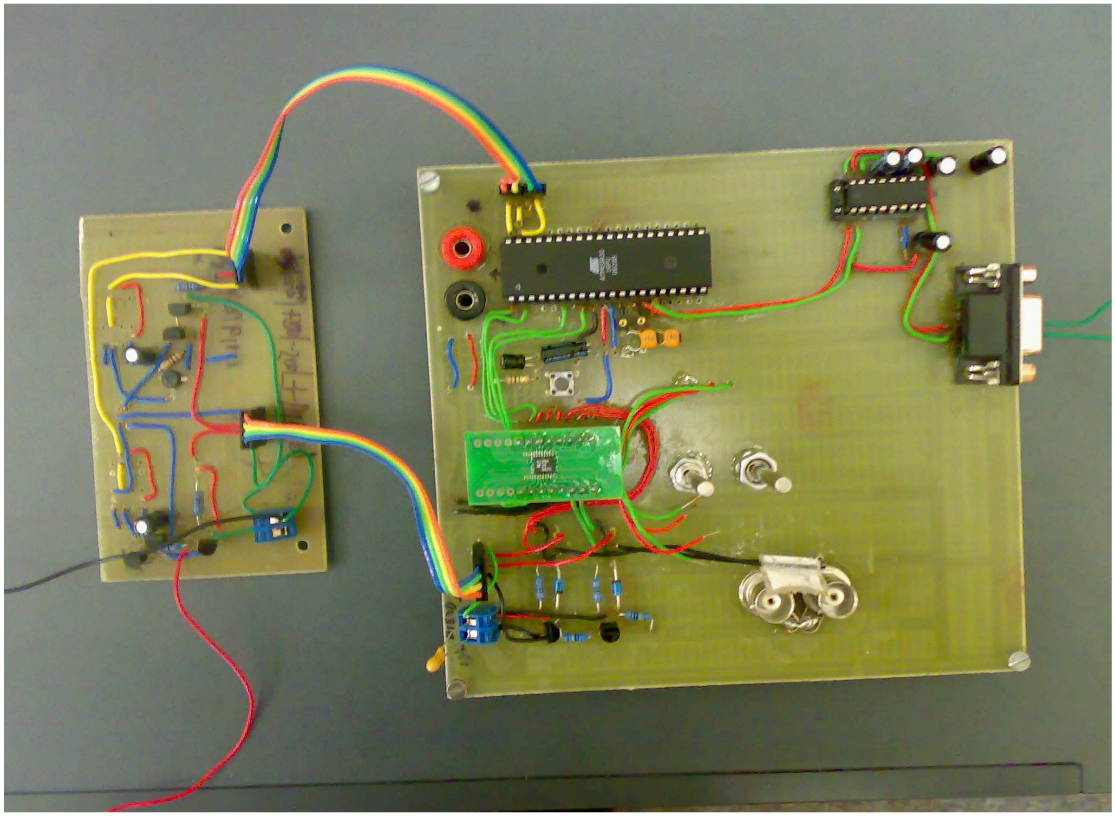
2.2.1 Verze 1

Na hlavní desce zapojen mikroprocesor ATmega32 v levé horní části, v pravé horní části je zapojen obvod Max 232 použitý k převodu úrovní +-10V EIA-232 na úrovně +5V 0V TTL a zpět. Dále je vidět zapojení digitálního potenciometru AD5235 a vpravo (obr 2-4) oba zdroje proudu, označené na schématu 2-5 červenou barvou.

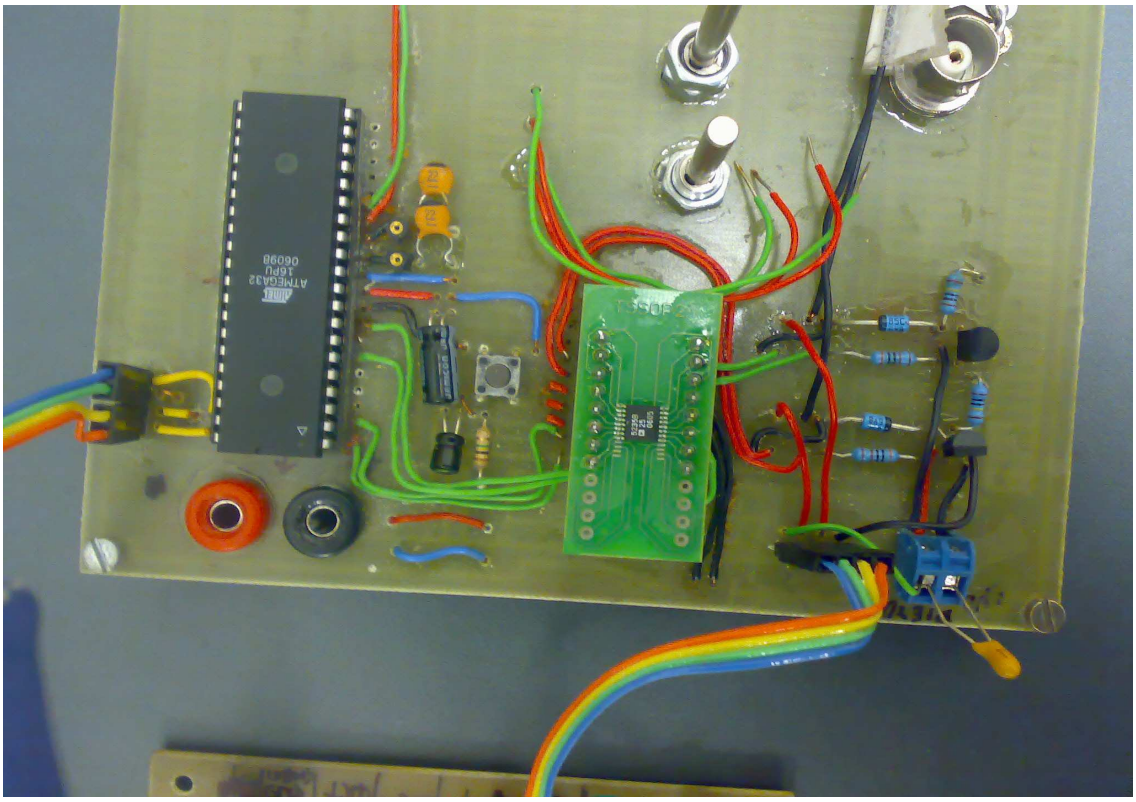
Digitální potenciometr je připojen na pozicích Rcon1 a R6. Jeho maximální hodnota je 25kOhm a je zapojen v sérii s 3kΩ odporem.

Dále je na vedlejší desce provedeno zapojení spínání, které je vidět na obrázku 2-3 a na schématu 2-5 zaznačeno zelenou barvou.

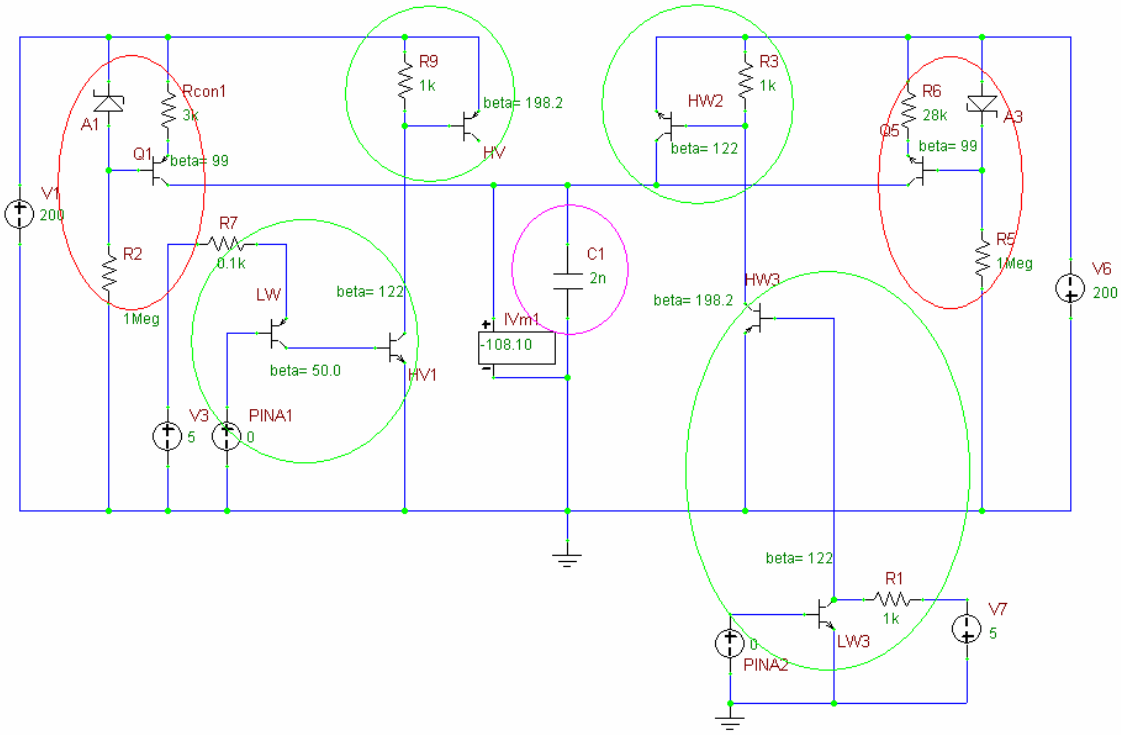
Fialovou barvou na schématu je vyznačen kondenzátor o hodnotě 2nF, který v návrhu obvodu nahrazuje piezokrystal.



Obrázek 2-3 Obvod zapojení mikroprocesoru a spínání



Obrázek 2-4 Detail zapojení mikroprocesoru a digitálního potenciometru



Obrázek 2-5 Schéma zapojení verze 1

2.2.2 Verze 2

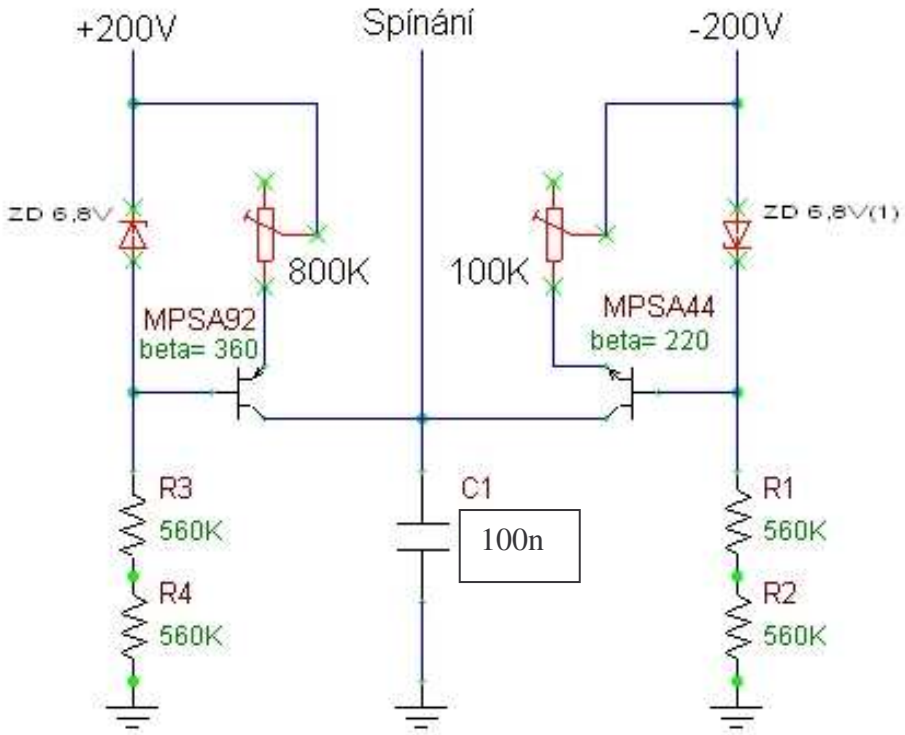
V druhé verzi byl návrh pozměněn z důvodu nemožnosti použití digitálního potenciometru pro řízení zdrojů proudu. Dále se při testování vybíjení na nepájivém poli doladili hodnoty rezistorů a typy tranzistorů. Bohužel se také zjistilo že rychlost vybíjení je několikanásobně vyšší než se původně předpokládalo a použitý mikroprocesor je příliš pomalý.

Testování spínacího obvodu bylo proto provedeno na kapacitoru o stonásobně větší hodnotě, aby bylo možné nastavit spínání s dostatečnou přesností.

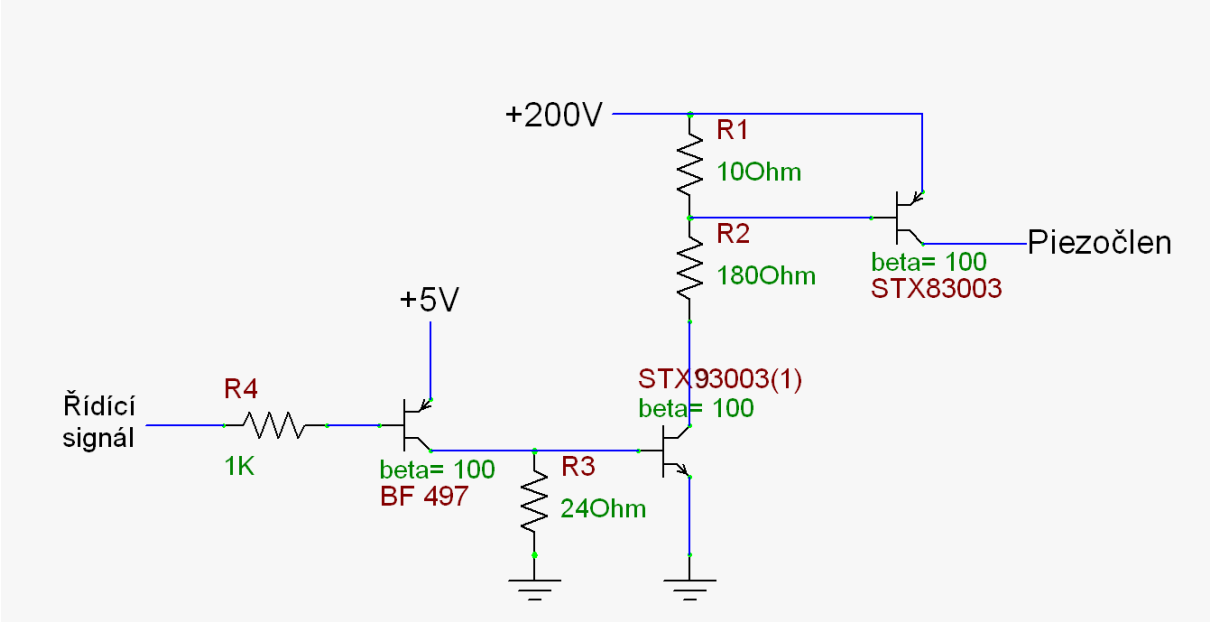
Na obrázku č.2-6 je zobrazeno upravené zapojení zdrojů proudu. Oproti původnímu návrhu se liší použitými tranzistory a nastavitelnými rezistory s mnohem vyššími hodnotami.

Obrázek 2-7 nám ukazuje otestované a funkční zapojení spínání levé části obvodu. Řídící signál je signál o logických úrovních +5V a 0V a je generován mikroprocesorem.

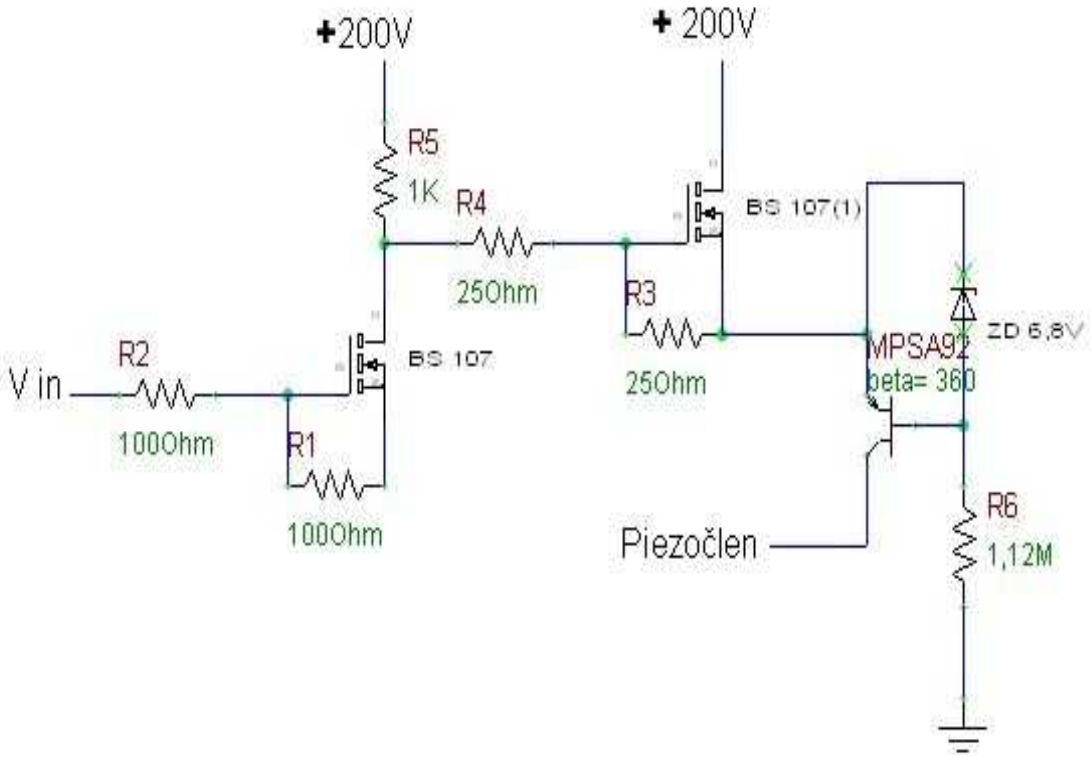
Náhrada za digitální potenciometr, který nemohl být použit, je zobrazena na obrázku č.2-8. Toto zapojení nebylo v době zpracování zprávy ještě otestováno. Předpokládá se, že vstupní napětí řídící velikost odporu bude generováno mikroprocesorem přes DA převodník.



Obrázek 2-6 Zdroje proudu



Obrázek 2-7 Spínání kladné polarity



Obrázek 2-8 Obvod nahrazující nastavitelný rezistor

2.3 Použité součástky

2.3.1 Mikroprocesor ATmega32

Srdce celého obvodu je mikroprocesor ATmega32 vyráběný firmou Atmel, patřící do rodiny 8-bitových riscových procesorů. Čip obsahuje 32k Bytovou flash paměť, sloužící jako paměť programu, dále 1k Byte paměti typu EEPROM a dále 2k Byte paměti SRAM. Dále může používat vnitřní oscilátor o frekvenci 8 MHz nebo vnější oscilátor až do frekvence 16MHz. Jeho součástí je rozhraní USART, SPI atd. Další důležité informace o procesoru lze nalézt v elektronické příloze nebo na stránkách firmy Atmel.

Primární úlohou procesoru je generování spínacího signálu pro zdroje proudu a nastavování hodnoty proudu dodávaného těmito zdroji. V budoucnu po otestování funkčnosti obvodu, bude procesor po zadání rychlosti posuvu již sám schopen vypočítat potřebné nastavení odporu a frekvence. V současné době je mnohem praktičtější tyto hodnoty zadat do procesoru ručně pomocí počítače a sériového rozhraní.

2.3.2 Převodník MAX232

Další obvod použitý v návrhu generátoru je převodník MAX232. Je to velmi rozšířený obvod, který slouží pro převod napěťových úrovní z TTL na úrovně používané ve standardu 232 a zpět. Zapojení je standardní, je uvedeno například v dokumentaci tohoto obvodu, který je v elektronické příloze. Tento obvod je napájen stejně jako mikroprocesor napětím +5V 0V. Ovšem standard 232 požaduje logické úrovně +-10V. Proto aby mohl pomocí tzv. Napěťové pumpy generovat požadované napětí, musí k němu být připojeno několik elektrolytických kondenzátorů.

2.3.3 Digitální potenciometr AD5235

Tento potenciometr byl použit v původním návrhu, bohužel se zjistilo, že je nevhodný pro aplikace s napětím vyšší jak 5,5V. Přesto bych se rád zmínil o jeho základních vlastnostech.

Tento potenciometr je vyráběn ve standardu TSSOP-16 a proto bylo třeba pro testování na nepájivém poli vyrobit redukci. Tuto redukci zhotovila firma

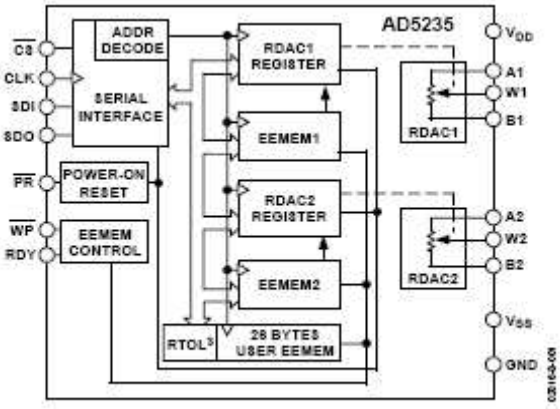
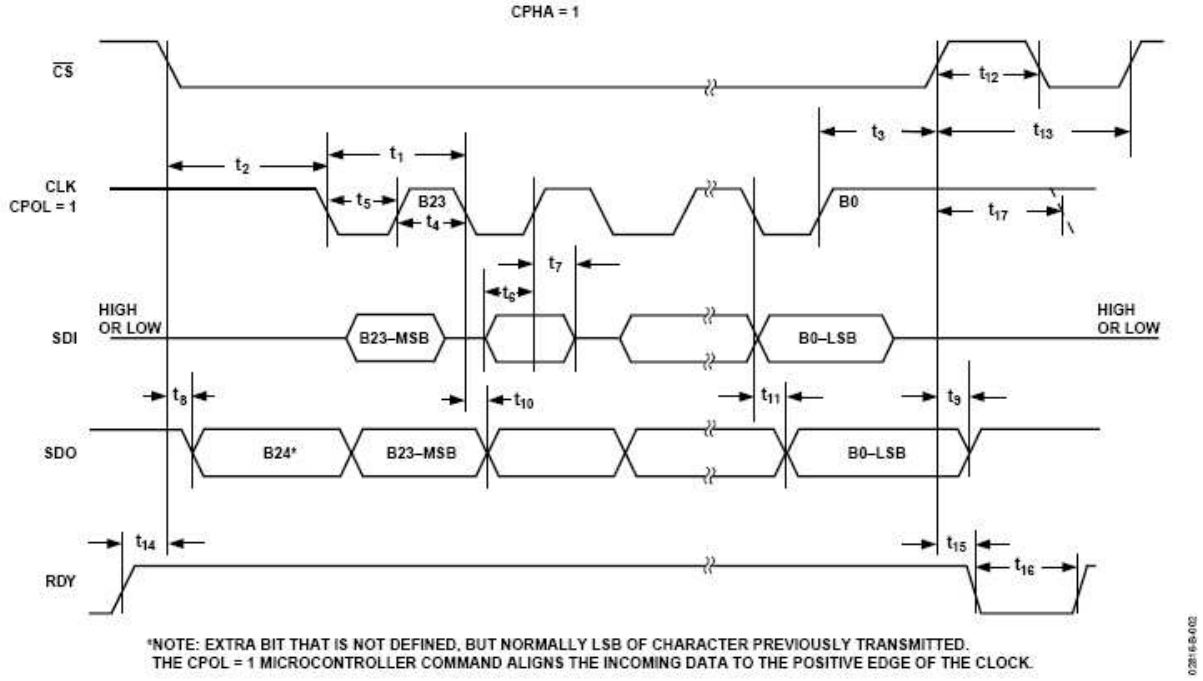


Figure 1.

PragoBoard podle návrhu, který jsem nakreslil v programu Eagle.

Na přiloženém obrázku Figure 1. je vidět schématické znázornění obsahu čipu.

Serial interface je synchronní sériové rozhraní určené ke komunikaci s mikroprocesorem dle následujícího časového diagramu.



Dále jsou zde registry RDAC1 a 2, podle kterých je nastaven výstup a obvod obsahuje pamět typu EEPROM o velikosti 26k Bytů.

2.3.4 Tranzistory STX

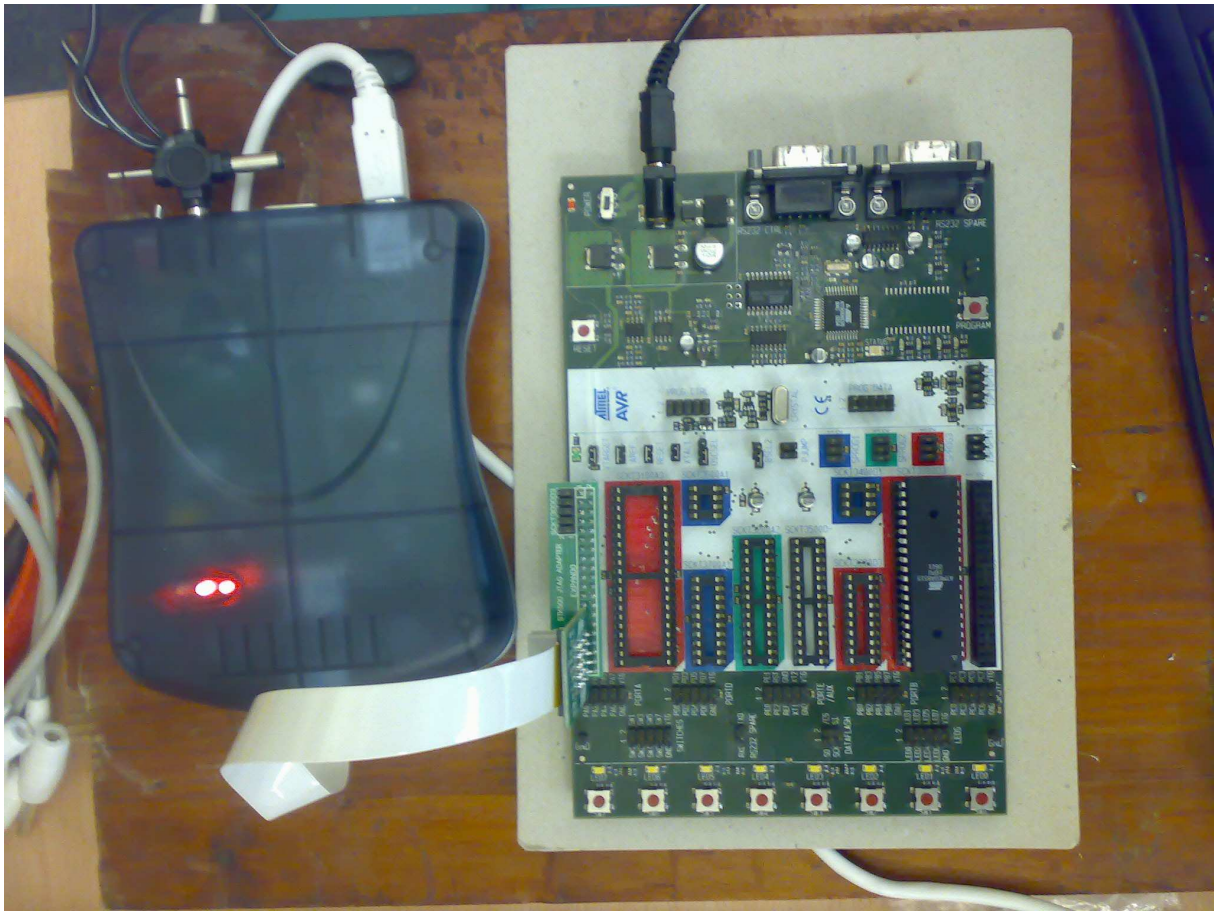
Jistou specialitou obvodu spínání jsou velmi rychlé vysokonapěťové tranzistory. Jde o typy STX83003 (PNP) a STX93003 (NPN), které zvládají rozdíl potenciálů 400V a jejich typická rozpínací doba je 0,2µs. Další informace a charakteristiky těchto tranzistorů lze nalézt v elektronické příloze.

Kapitola 3 Program pro mikroprocesor

3.1 Nástroje pro programování mikroprocesoru

Program je vytvořen v AVRStudiu v kódu AVR Assembleru. Assembler jsem upřednostnil před jazykem C kvůli přesné spočitatelnosti času provádění jednotlivých kritických úseků instrukcí.

K programování jsem použil univerzální programovací jednotku pro procesory AVR, STK-500, s připojenou jednotkou JTAGICE mk-2, která umožňuje odladění programu nahraného v mikroprocesoru. Zapojení lze vidět na obrázku č.3-1.



Obrázek 3-1Přípravek STK 500 a JTAGICE mk-2

3.2 Program pro zapojení dle verze 1

Program byl složen z částí pro komunikaci po sériovém portu, komunikaci po SPI rozhraní s digitálním potenciometrem a části pro výpočet intervalu nabíjení piezočlenu v závislosti na zvolené frekvenci.

3.2.1 UART rozhraní

Ze strany mikroprocesoru byl použit Hardwarový UART nastavený na přenosovou rychlost 9600 Baudů. Nastavení v AvrAssembler vypadá následovně:

```
ldi temp2,$0
;ldi temp,$67;nastaveni baud rate na 9600 pri krystalu 16MHz
ldi temp,$33;nastaveni baud rate na 9600 pri krystalu 8MHz
call USART_Init

USART_Init:
; Set baud rate
out UBRRH, temp2
out UBRRL, temp
; Enable receiver and transmitter
ldi r16, (1<<RXEN)|(1<<TXEN)|(1<<RXCIE)| (1<<TXCIE)
out UCSRB,r16
; Set frame format: 8data, 2stop bit
ldi r16, (1<<URSEL)|(1<<USBS)|(3<<UCSZ0)
out UCSRC,r16
ret
```

Příjem dat je obsluhován v proceduře obsluhy přerušení ukončení příjmu dat. Přijatá data lze nalézt v registru UDR (UART data register). Ten se používá jak pro příjem dat tak pro vysílání, ovšem při zápisu do tohoto registru (vysílání) jsou data směrována do jiné části registru než když z registru čteme (příjem dat).

Ze strany počítače byl ke komunikaci použit FreeWare program Herkules umožňující jednoduché nastavení com portu a dále jednoduše odesílat jak Ascii znaků tak dat v hex formátu. Program je přiložen na cd.

3.2.2 Obsluha přerušení ukončení příjmu dat

Zde se vyhodnotí přijatý znak a pokud se shoduje se znakem příkazu, mikroprocesor uloží dalších osm bitů na pozici určenou daným příkazem, zavolá se procedura pro spočítání nastavení potenciometru a odeslání dat po SPI rozhraní. Registr UDR je vstupně/výstupní a proto s k jeho obsluze používají instrukce in a out.

3.2.3 SPI rozhraní

Nastavení SPI rozhraní je následující:

SPCR register mikroprocesoru je nastaven na hodnotu 5F hex. To znamená že:

Rychlost přenosu je nastavena na $f_{osc}/128$.
Poloha hodin v klidu je v log 1.
Data jsou čtena na náběžnou hranu.
MSB je vysílán jako první.
SPI rozhraní je v módu Master.
SPI zapnuto.

Dále je nutno při přenosu příkazu potřeba nastavit RDY bit na log 1 CSEL na log 0.

Pak již jen stačí zapsat data do registru SPDR a mikroprocesor se postará o zbytek. Při ukončení přenosu je nastaven osmý bit registru SPSR na log 1. Po jeho přečtení, je nastaven opět do log 0 až do ukončení dalšího přenosu.

Příkazy pro digitální potenciometr lze vzhledat v přiložené dokumentaci. Zde popíši jen ten, který byl použit v programu. Jde o příkaz zápisu do registru RDAC1 a 2 podle kterých je nastavena hodnota.

Tento příkaz se skládá z:

První čtyři bity jsou využity pro identifikaci instrukce (v našem případě 1011)

Z dalších čtyřech bitů je pro tuto instrukci využit jen poslední pro určení registru (RDAC1-0000 a RDAC2-0001)

Dalších 16 bitů udává hodnotu v rozmezí 1-1024, která bude zapsána do registru.

3.2.4 Procedura pro výpočet intervalu nabíjení

Tato procedura nebyla již implementována. Jejím základem by byla skutečnost, že piezokrystal se chová jako kapacitor o známé hodnotě. Dále bychom znali hodnotu proudu přitékajícího do piezokrystalu podle nastavení digitálních potenciometrů. Podle vzorce $t = \frac{U * C}{I}$ by bylo již snadné dopočítat tento časový údaj.

3.3 Program pro zapojení dle verze 2

Stejně jako pro verzi 1 obsahuje část zajišťující komunikaci po rozhraní UART. A dále dva softwarové časovače navržené tak, aby vyhovovali požadavkům obvodu. Je také implementována sada příkazů, za pomoci kterých se nastavují parametry časovačů.

3.3.1 Příkazy a časovače

Časovače jsou implementovány pomocí tří vnořených forcyklů v případě délky periody, v případě délky pulsu postačuje jeden cyklus.

Následující úsek kódu je časovač použitý pro určení délky periody. Minimální čas je 1,6µs a s každým zvýšením celkového počtu cyklů o 1 se doba zvýší o 0,4µs. Tyto časy byli změřeny při použití 8MHz oscilátoru.

```
delayH:

        mov coarse,coarseH
        mov medium,mediumH
        mov fine,fineH
againH:                         dec fine
                        brne againH
                dec medium
                brne againH
        dec coarse
        brne againH
ret
```

V následující tabulce jsou uvedeny příkazy pro zadávání parametrů generovaného průběhu. Písmeno je posíláno ve formě Ascii kódu a hodnota hexadecimálně.

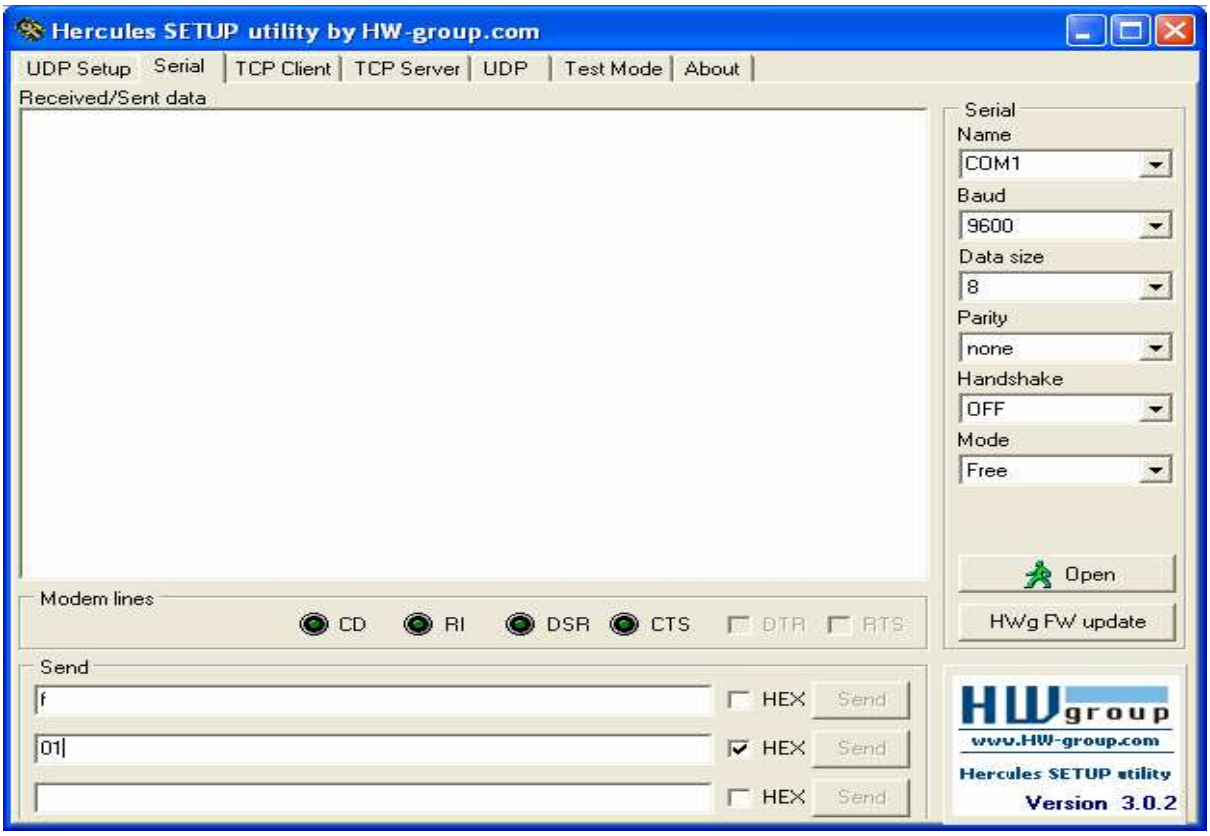
Příkaz	Popis
f<hodnota>	Nastavení délky vybíjecího pulsu. Povolené hodnoty jsou 01-FF. Přepočet na čas je dle vzorce $t = 1 + 0{,}4 * hodnota[\mu s]$.
F< hodnota >	Nastavení hodnoty fine pro určení délky periody. Povolené hodnoty jsou 01-FF.
M< hodnota >	Nastavení hodnoty medium pro určení délky periody. Povolené hodnoty jsou 01-FF.
C< hodnota >	Nastavení hodnoty coarse pro určení délky periody. Povolené hodnoty jsou 01-FF.

Délka periody je určena vzorcem

$$T = t + [1{,}6 + (coarse * 255^2 + mediu * 255 + fine) * 0{,}4]$$

3.3.2 Komunikace ze strany PC

Pro komunikaci po sériové lince byl použit FreeWare program Herkules. Umožňuje jednoduché nastaveni sériového portu a bezproblémovou komunikaci. Následující screenshot ukazuje základní nastavení.



Obrázek 3-2 Nastavení programu Herkules

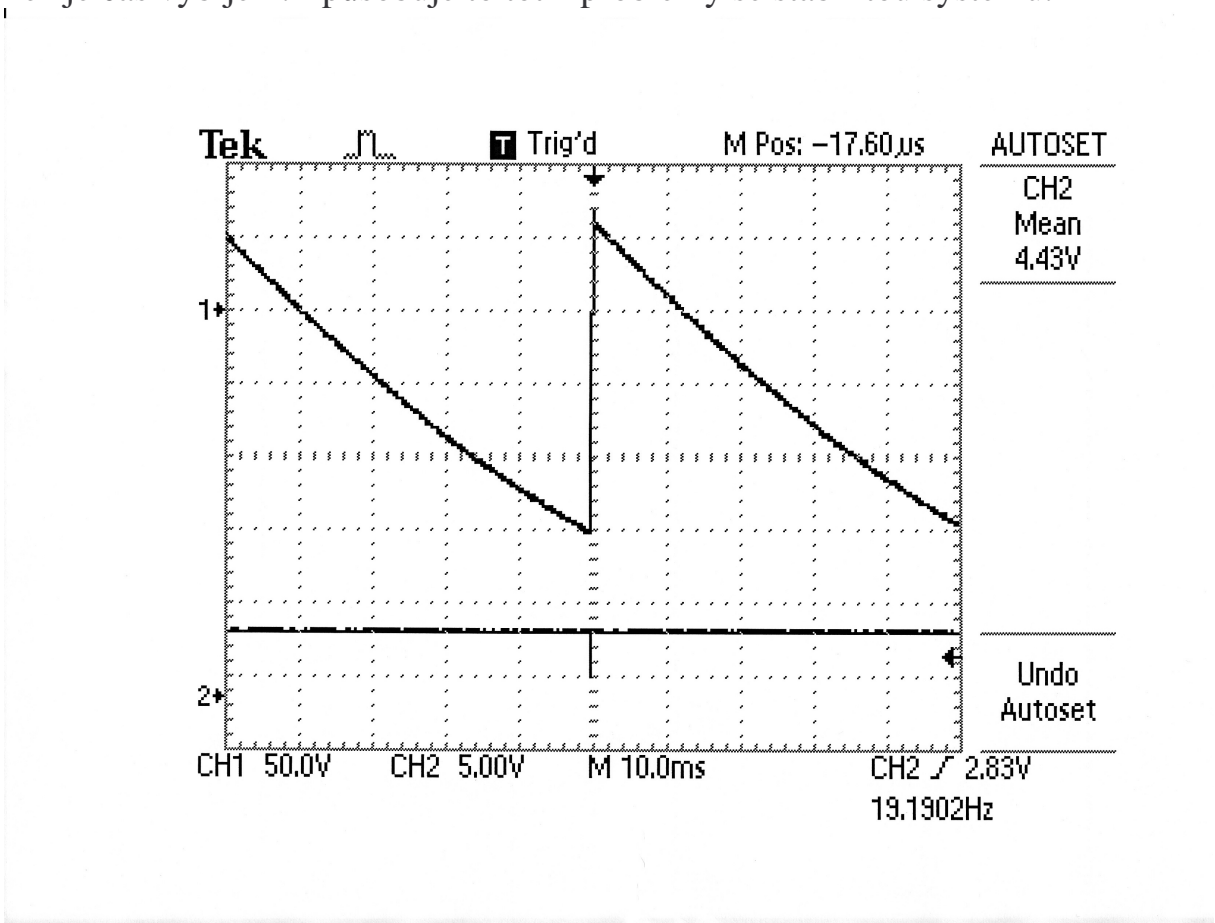
Kapitola 4 Testování

Přiblížení k požadovaným výsledkům přineslo testování až druhého zapojení. Při těchto testech byly součástky umístěny v nepájivém poli a proto se dá očekávat ještě zlepšení kvality řídícího signálu i generovaného průběhu napětí.

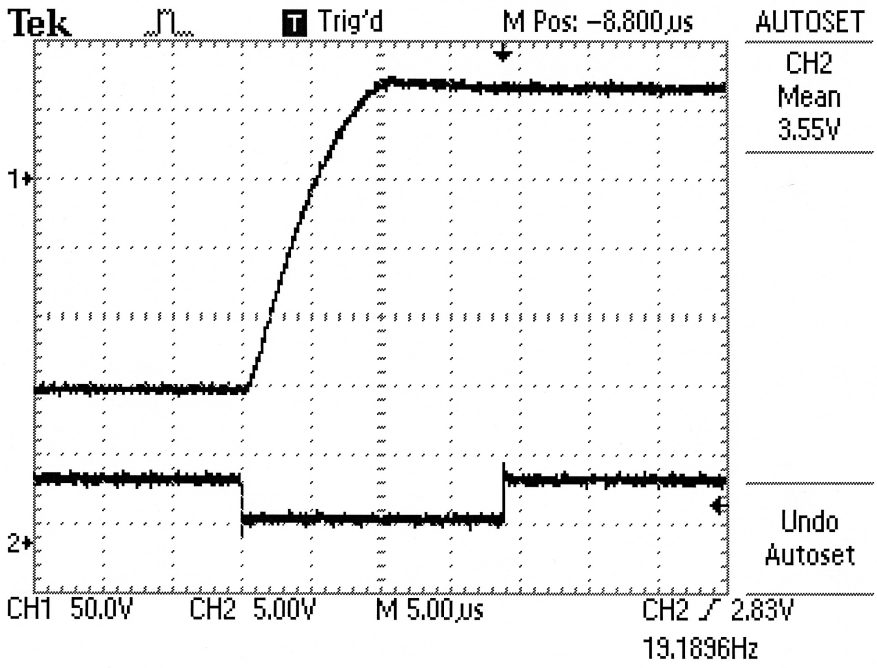
Dále bylo nutné zvolit mnohonásobně větší kapacitor něž ten, který se přibližoval charakteristice piezokrystalu, protože doba vybíjení je mnohem menší než se původně očekávalo a překračuje možnosti zvoleného mikroprocesoru.

Na obrázku 4-1 je zachycen průběh generovaného napětí na kapacitoru. Blíží se k nám požadovanému pilovitému průběhu, i když je vidět mírná nelinearita. Ta ovšem nebude mít vliv na funkci motoru, pouze se bude muset odrazit ve výpočtu času nabíjení.

Obrázek 4-2 zachycuje detail vybíjení a dole je zobrazen spouštěcí signál z mikroprocesoru. Ten bude později nastavovat tak aby nebyl o mnoho delší než je čas vybíjení. Způsobuje to totiž problémy se stabilitou systému.



Obrázek 4-1 Generovaný průběh napětí na kapacitoru



Obrázek 4-2 Detail vybíjení kapacitoru

Kapitola 5 Závěr

5.1 Dosažené výsledky

Na začátku práce bylo třeba vytvořit návrh obvodu a vybrat mikroprocesor, který řídí spínání zdrojů proudu. Obvod navrhoval vedoucí bakalářské práce Ing. Martin Špiller a výběr mikroprocesoru jsem provedl já.

Dále po dokončení první verze návrhu bylo třeba vytvořit redukce pro SMD potenciometry použité v obvodu, aby bylo možné otestovat zapojení na nepájivém poli.

Na univerzálním pájecím poli jsem připravil zapojení pro řídící procesor, obsahující externí krystal o hodnotě 16MHz, dále rozhraní RS232 pro komunikaci s počítačem.

Dále jsem připravil kód v AVR Studiu komunikující s digitálním potenciometrem za pomoci rozhraní SPI.

Po otestování funkčnosti navržených zdrojů proudu na nepájivém poli jsem je napájel na univerzální pájecí pole. Spínání bylo vytvořeno na oddělenou desku připojitelnou pomocí dvou čtyřžilových kabelů.

Zprovoznil jsem komunikaci procesoru s počítačem pomocí rozhraní RS232 a připravil program pro spínání zdrojů proudu.

Při testování první verze zapojení jsem zjistil, že použití digitální potenciometru je bohužel zcela nevhodný pro dané zapojení. Na rezistoru digitální potenciometru totiž nelze použít napětí vyšší než jeho napájecí napětí. Proto byli nahrazeny analogovými potenciometry. Toto zapojení však již nebylo dále testováno z důvodu nepraktičnosti používání analogových potenciometrů.

Při testování druhé verze návrhu jsme narazili na další překážku v podobě rychlosti mikroprocesoru. Pro tvorbu spouštěcího pulzu bude třeba aby se minimální délka pohybovala pod hranicí 100ns a hodnota jednoho dílku byla 10-20ns. Minimální hodnoty půjde dosáhnout použitím procesoru umožňující 20MHz rychlost a vnitřního čítače nastaveného na maximální rychlost (tzn. rychlost mikroprocesoru). Ovšem dostatečně jemné rozlišení takto dosáhnout nelze, protože při frekvenci 20MHz je minimální rozlišení 50ns.

20MHz je ale maximální frekvence pro 8-bitové procesory od firmy Atmel. My bychom ovšem potřebovali alespoň 50MHz.

Bude se tedy nejspíše najít jiné řešení.

Nabízelo se řešení použít obvod 555, který funguje na principu RC členu. Tady se naráží na problém, že spouštěcí impuls musí být kratší než impuls vygenerovaný a navíc u standardního typu tohoto časovače je náběžná a sestupná hrana na výstupu dlohá 100ns.

5.2 Pokračování projektu

Problémy, které aktuálně vyvstávají a jsou potřeba vyřešit, jsou následující :

- Navržení a otestování spínání pro opačnou polaritu
- Otestování funkčnosti náhrady za potenciometry
- Výběr a zprovoznění DA převodníků pro řízení zdrojů proudu
- Vyřešení problému s rychlostí spínání (hledání rychlejšího mikroprocesoru nebude nejspíš správný směr)
- Otestování na reálném piezomotoru a změření charakteristik
- Tvorba spínání pro složitější průběh pily typu posicast pokud bude třeba potlačit oscilace
- Otestování složitějšího spínání
- Tvorba algoritmu pro výpočet údajů pro zadanou rychlost a směr posunu
- Atd.

Hlavní problém bude s tvorbou dostatečně rychlého signálu pro sepnutí. Potýkáme se s ním již od začátku a zatím nebylo nalezeno uspokojivé řešení.

Zbytek kroků je spíše otázkou času a trpělivosti.

5.3 Osobní zhodnocení

Celá tato práce byla složena s mnoha úkolů, od výběru vhodných součástek, přes návrh plošných spojů a jejich následnou realizaci až po tvorbu řídícího programu pro mikroprocesor. Bylo to velmi důkladné otestování mých znalostí a schopností a mnoho nových věcí bylo třeba se ještě naučit.

Práci na projektu jsem přerušil ve chvíli, kdy se přechází od fáze vývoje k fázi ladění a testování. Je třeba dodat že první výsledky jsou uspokojivé ovšem ještě nebyl generátor testován na reálném motoru.

5.4 Použitá literatura a CD přílohy

[1] Odborný článek http://www.odbornecasopisy.cz/automa/2007/au040708.htm

[2] David Matoušek, Práce s mikrokontroléry Atmel AVR, Ben, 2006

[3] Vladimír Váňa, Mikrokontroléry ATLMEL AVR – assembler, Ben, 2003

[4] Vladimír Váňa, Mikrokontroléry ATLMEL AVR – programování v jazyce C, Ben, 2003

[5] Arvid Bergander, Control, Wear Testing & Integration of Stisk-slip Micropositioning, EPFL, 2003

[6] William Singhose, Command Generation for Flexible Systems, MIT, 1997

CD:

[1] Herkules.exe – Freeware program nejen pro komunikaci po sériovém rozhraní

[2] Piezo.asm – Aktuální verze programu pro mikroprocesor

[3] mega32.pdf – Dokumentace k mikroprocesoru

[4] AD5235.pdf – Dokumentace k digitálnímu potenciometru

[5] max232.pdf – Dokumentace k obvodu max232

[6] stx83003.pdf – Dokumentace k tranzistoru stx83003