

UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

SELEKTIVNÍ VOLTMETR NA BÁZI ČÍSLICOVÉHO FILTRU

Josef Jelínek

Bakalářská práce
2020

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Josef Jelínek**
Osobní číslo: **I17299**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Řízení procesů**
Téma práce: **Selektivní voltmetr na bázi číslicového filtru**
Zadávající katedra: **Katedra řízení procesů**

Zásady pro vypracování

Úkolem je navrhnout selektivní voltmetr užívající přeladitelný číslicový filtr typu pásmová propust pro telefonní pásmo kmitočtů 500 Hz až 3000 Hz.
Po analogovém preprocesingu se měřený signál převede do číslicového tvaru, filtruje se úzkopásmovým číslicovým filtrem typu pásmová propust a napětí se zobrazí na displeji. Vlastní filtr je realizován mikroprocesorem, přeladování se děje z klávesnice, naladěný kmitočet zobrazuje displej. Maximální vstupní úroveň se předpokládá + 4 dBm.

Rozsah pracovní zprávy:
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

SMĚKAL, Zdeněk, SYSEL, Petr. Signálové procesory. Praha: Sdělovací technika, 2006. ISBN 80-86645-08-8.
MATOUŠEK, David, BRTNÍK, Bohumil. Aplikace algoritmů číslicového zpracování signálů 1. Praha: BEN – technická literatura, 2014. ISBN 978-80-7300-478-1.
SMĚKAL, Zdeněk, VÍCH, Robert. Zpracování signálů pomocí signálových procesorů. Praha: Radix, 1998. ISBN 80-86031-18-7.
MATOUŠEK, D.: Aplikace mikrokontrolérů ATmega644. BEN-technická literatura. Praha, 2013. ISBN 978-80-7300-492-7

Vedoucí bakalářské práce: **Ing. Bohumil Brtník, Dr.**
Katedra elektrotechniky

Datum zadání bakalářské práce: **17. prosince 2019**
Termín odevzdání bakalářské práce: **7. května 2020**



L.S.

Ing. Zdeněk Němec, Ph.D.
děkan

Ing. Daniel Honc, Ph.D.
vedoucí katedry

V Pardubicích dne 20. ledna 2020

Prohlášení

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne

Josef Jelínek

Poděkování

Velmi děkuji Ing. Bohumilu Brtníkovi, Dr. za technickou a pedagogickou pomoc v celém průběhu zpracování mé bakalářské práce.

V Pardubicích dne

Josef Jelínek

ANOTACE

Bakalářská práce se zabývá návrhem selektivního voltmetru na bázi číslicového filtru typu pásmová propust. Vstupní analogové napětí se převádí do číslicového tvaru. Po průchodu filtrem se zobrazuje na LCD displeji střídavě se zvoleným kmitočtem. Součástí práce je deska plošných spojů. Hlavní částí je mikroprocesor ATmega644, pomocí kterého je realizován číslicový filtr. Software pro mikroprocesor je napsán v jazyku C.

KLÍČOVÁ SLOVA

číslicový filtr, selektivní voltmetr, ATmega644, Butterworthova aproximace.

TITLE

SELECTIVE VOLTMETER BASED ON DIGITAL FILTER

ANNOTATION

This bachelor thesis deals with selective voltmeter based on digital band-pass filter. Input analog voltage is converted to digital form. After passing filter is voltage shown on LCD display and alternately is shown selected frequency. Bachelor thesis includes a printed circuit board. Main part of board is microprocessor ATmega644, that implements digital filter. The microprocessor software is written in C.

KEYWORDS

Digital filter, Selective voltmeter, ATmega644, Butterworth approximation.

OBSAH

| | |
|--|----|
| OBSAH | 7 |
| SEZNAM ZKRATEK A ZNAČEK | 9 |
| SEZNAM SYMBOLŮ PROMĚNNÝCH VELIČIN A FUNKCÍ..... | 11 |
| SEZNAM ILUSTRACÍ | 13 |
| SEZNAM TABULEK..... | 15 |
| ÚVOD | 16 |
| 1 TEORETICKÁ ČÁST | 17 |
| 1.1 SELEKTIVNÍ VOLTMETRY | 17 |
| 1.2 ELEKTRICKÉ FILTRY | 18 |
| 1.2.1 Znázornění činnosti analogového filtru typu dolní propust..... | 20 |
| 1.2.2 Znázornění činnosti číslicového filtru typu dolní propust..... | 21 |
| 1.2.3 Popis činnosti filtrů..... | 23 |
| 1.2.4 Obecná rovnice pro realizaci číslicového filtru | 25 |
| 1.2.5 Stabilita číslicového filtru..... | 27 |
| 1.2.6 Závislost frekvenční charakteristiky na poloze pólů | 29 |
| 1.2.7 Impulsní charakteristika číslicového filtru | 30 |
| 1.3 REALIZACE ČÍSLICOVÝCH FILTRŮ | 31 |
| 1.3.1 Metoda rozmístění nulových bodů a pólů..... | 32 |
| 1.3.2 Metoda impulsní invariance..... | 33 |
| 1.3.3 Metoda bilineární transformace | 34 |
| 2 VÝBĚR KOMPONENT | 38 |
| 2.1 MIKROPOČÍTAČ | 38 |
| 2.1.1 Mikropočítač ATmega644 | 39 |
| 2.2 LCD DISPLEJ | 40 |
| 2.3 NAPÁJENÍ OBVODU | 40 |
| 2.4 A/Č PŘEVODNÍK..... | 41 |
| 2.4.1 Popis činnosti A/Č převodníku | 42 |
| 2.4.2 Režimy startu převodu | 44 |
| 2.4.3 Předdělička A/Č převodníku..... | 44 |
| 2.4.4 Uložení výsledku převodu | 45 |
| 2.4.5 Změna kanálu..... | 46 |
| 2.4.6 Registry..... | 46 |

| | | |
|-------|--|----|
| 2.5 | JEDNOTKA USART | 47 |
| 2.6 | JEDNOTKA TWI..... | 49 |
| 2.7 | SÉRIOVÝ KANÁL SPI | 50 |
| 3 | REALIZACE HARDWARU..... | 52 |
| 3.1 | NÁVRH DPS..... | 56 |
| 4 | REALIZACE SOFTWARE | 60 |
| 4.1 | NÁVRH ČÍSLICOVÉHO FILTRU..... | 60 |
| 4.1.1 | Výpočet filtru s kmitočtem f_1 | 61 |
| 4.1.2 | Výpočet filtru s kmitočtem f_2 | 63 |
| 4.1.3 | Výpočet filtru s kmitočtem f_3 | 65 |
| 4.1.4 | Výpočet filtru s kmitočtem f_4 | 67 |
| 5 | ZÁVĚR | 73 |
| | POUŽITÁ LITERATURA | 74 |
| | PŘÍLOHY | 75 |

SEZNAM ZKRATEK A ZNAČEK

| | |
|--------|---|
| ADCH | datový registr A/Č převodníku (horní bajt) |
| ADCL | datový registr A/Č převodníku (dolní bajt) |
| ADMUX | řídící registr multiplexeru A/Č převodníku |
| ADCSR | řídící a stavový registr A/Č převodníku |
| A/Č | analogově-číslicový |
| CLK | hodinový signál |
| CTS | vstupní vodič pro řízení přenosu |
| Č/A | číslicově-analogový |
| DP | dolní propust |
| DPS | deska plošných spojů |
| E | signál pro synchronizaci |
| EEPROM | elektronicky vymazatelná paměť pouze pro čtení |
| FIR | filtr s konečnou impulsní odezvou |
| FLASH | elektricky programovatelná paměť s libovolným přístupem |
| HDO | hromadné dálkové ovládání |
| HP | horní propust |
| IIR | filtr s nekonečnou impulsní odezvou |
| LCD | displej z kapalných krystalů |
| LED | dioda emitující světlo |
| MISO | master vstup, slave výstup |
| MOSI | master výstup, slave vstup |
| NDP | normovaná dolní propust |
| PC | osobní počítač |
| PP | pásmová propust |
| PWM | pulzně šířková modulace |
| PZ | pásmová zádrž |
| RAM | paměť s náhodným přístupem |
| RISC | redukovaná instrukční sada |
| RS | signál rozlišující příkazy od dat |
| RS-232 | sériová linka |
| RTS | výstupní vodič pro řízení přenosu |
| Rx | vysílání dat |

| | |
|-------|--|
| R/W | signál určující směr komunikace |
| SCLK | hodinový signál |
| SPI | sériové periferní rozhraní |
| SRAM | statická paměť |
| TTL | tranzistorově-tranzistorová logika |
| TWI | dvoudrátové rozhraní |
| Tx | přijímání dat |
| UDR | datový registr USART |
| USART | univerzální synchronní / asynchronní přijímač / vysílač (sériové rozhraní) |
| USB | univerzální sériová sběrnice |
| μP | mikroprocesor |

SEZNAM SYMBOLŮ PROMĚNNÝCH VELIČIN A FUNKCÍ

| | |
|-------------|--|
| A | amplituda, V |
| A_c | propustné pásmo |
| $AGND$ | analogové uzemnění |
| A_N | napět'ové zesílení zesilovače |
| A_{REF} | referenční napětí, V |
| A_s | útlum filtru |
| A_{UCC} | napájecí napětí |
| d | útlumový činitel |
| f_{ADC} | hodinový kmitočet, Hz |
| f_c | mezní kmitočet, Hz |
| f_{DP} | frekvence dolní propusti, Hz |
| f_{HP} | frekvence horní propusti, Hz |
| f_s | mezní kmitočet, Hz |
| f_{VZ} | vzorkovací kmitočet, Hz |
| $F(s)$ | přenos filtru |
| $F(z)$ | přenosová funkce filtru |
| $F(\omega)$ | frekvenční charakteristika |
| G | zisk, dB |
| GND | uzemnění |
| $g(t)$ | přechodová charakteristika |
| $h(t)$ | impulsní charakteristika |
| k | činitel selektivity filtru |
| $l(t)$ | jednotkový skok |
| p | pól přenosu |
| $P(z)$ | pomocná proměnná |
| Q | činitel jakosti |
| R_{IN} | vstupní odpor, Ω |
| R_{REF} | vstupní odpor pro A_{REF} , Ω |
| s | symbolický kmitočet |
| t | spojitý čas, s |
| t_{CONV} | doba převodu, s |
| T_{VZ} | vzorkovací perioda, s |

| | |
|----------------|---|
| U_{CC} | napětí, V |
| U_{NEG} | napětí invertujícího vstupu, V |
| U_{POS} | napětí neinvertujícího vstupu, V |
| U_{VST} | vstupní napětí, V |
| $u(n)$ | diskrétní napětí, V |
| $u(t)$ | spojité napětí, V |
| $v(t)$ | signál přibližující se obdélníkovým impulsům |
| $x(n)$ | vstupní vzorek |
| $y(n)$ | výstupní vzorek |
| $\Delta\Omega$ | šířka pásma |
| $\delta(t)$ | Diracův impuls |
| Ω | úhlový kmitočet, $\text{rad}\cdot\text{s}^{-1}$ |
| Ω_c | mezní úhlový kmitočet, $\text{rad}\cdot\text{s}^{-1}$ |
| ω | úhlová frekvence, $\text{rad}\cdot\text{s}^{-1}$ |
| ω_c | mezní úhlový kmitočet, $\text{rad}\cdot\text{s}^{-1}$ |
| ω_s | mezní úhlový kmitočet, $\text{rad}\cdot\text{s}^{-1}$ |

SEZNAM ILUSTRACÍ

| | |
|--|----|
| Obr. 1.1 – Blokové schéma nízkofrekvenčního selektivního voltmetru..... | 17 |
| Obr. 1.2 – Nízkofrekvenční selektivní zesilovač | 17 |
| Obr. 1.3 – Blokové schéma heterodynního mikrovoltmetru s jedním směřováním..... | 18 |
| Obr. 1.4 – Útlumové charakteristiky jednotlivých filtrů (Brtník, 2014)..... | 19 |
| Obr. 1.5 – Přenosové charakteristiky filtrů a odezvy filtrů na jednotkový skok pro NDP (Brtník, 2014)..... | 19 |
| Obr. 1.6 – Téměř obdélníkové impulsy rozloženy na dva sinusové signály (Brtník, Matoušek, 2012)..... | 20 |
| Obr. 1.7 – Časové průběhy a spektrum na vstupu a výstupu dolní propusti (Brtník, Matoušek, 2012)..... | 21 |
| Obr. 1.8 – Názorný příklad činnosti číslicového filtru typu dolní propust (Brtník, Matoušek, 2010)..... | 22 |
| Obr. 1.9 – Blokové schéma filtru typu dolní propust (Brtník, Matoušek, 2010)..... | 23 |
| Obr. 1.10 – Graf přenosové funkce $F(\omega) = 2 \cdot e^{-j\omega \cdot T_{vz}/2} \cdot \cos \omega \cdot \tau / 2$ dolní propusti (Brtník, Matoušek, 2014) | 24 |
| Obr. 1.11 – Vstupní antialiasingový a výstupní rekonstrukční analogový filtr u číslicového filtru (Brtník, Matoušek, 2010)..... | 25 |
| Obr. 1.12 – Blokové schéma filtru IIR o m-článcích (Brtník, Matoušek, 2010)..... | 26 |
| Obr. 1.13 – K otázce stability filtru IIR (Brtník, Matoušek, 2010) | 28 |
| Obr. 1.14 – K otázce stability filtru FIR (Brtník, Matoušek, 2010) | 29 |
| Obr. 1.15 – Vliv nulových bodů a pólů na průběh kmitočtové charakteristiky $F(s)$ (Brtník, Matoušek, 2014) | 30 |
| Obr. 1.16 – Transformace roviny p na rovinu z (Smékal, 1993)..... | 32 |
| Obr. 1.17 – Rozložení pólů Butterworthových polynomů na kružnici pro různé řády (Brtník, Matoušek, 2014). | 36 |
| Obr. 1.18 – K Butterworthovu polynomu druhého řádu (Brtník, Matoušek, 2014)..... | 37 |
| Obr. 2.1 – LCD alfanumerický displej | 40 |
| Obr. 2.2 – Blokové schéma A/D převodníku (Matoušek, 2013) | 43 |
| Obr. 2.3 – Spouštěcí logika A/D převodníku (Matoušek, 2013) | 44 |
| Obr. 2.4 – Předdělička pro A/D převodník (Brtník, Matoušek, 2010)..... | 45 |
| Obr. 2.5 – K registru ADMUX (Váňa, 2003)..... | 47 |
| Obr. 2.6 – K registru ADCSRA (Váňa, 2003)..... | 47 |

| | |
|--|----|
| Obr. 2.7 – Blokové schéma jednotky USART (Hrbáček, 2002) | 48 |
| Obr. 2.8 – Časování přenosu pomocí USART (src.athaj.cz, nedatováno) | 49 |
| Obr. 2.9 – Ukázka zapojení sběrnice TWI (zavavov.cz, 2014) | 50 |
| Obr. 2.10 – Průběh přenosu dat na linkách SDA a SCL (zavavov.cz, 2014) | 50 |
| Obr. 2.11 – Princip komunikace po sběrnici SPI (root.cz, 2008) | 51 |
| Obr. 2.12 – Průběh přenosu dat po sběrnici SPI (root.cz, 2008) | 51 |
| Obr. 3.1 – Schéma realizovaného obvodu | 54 |
| Obr. 3.2 – Schéma se dvěma fyzickými displeji..... | 55 |
| Obr. 3.3 – DPS selektivního voltmetru..... | 56 |
| Obr. 3.4 – DPS selektivního voltmetru s vylitým polygonem..... | 57 |
| Obr. 3.5 – DPS LCD displeje | 57 |
| Obr. 3.6 – DPS LCD displeje s vylitým polygonem | 58 |
| Obr. 3.7 – Osazená deska selektivního voltmetru (strana SMD součástek)..... | 58 |
| Obr. 3.8 – Osazená deska LCD displeje (strana SMD součástek)..... | 59 |
| Obr. 3.9 – Osazená deska selektivního voltmetru s LCD displejem | 59 |
| Obr. 4.1 – Blokové schéma filtru s kmitočtem f_1 | 63 |
| Obr. 4.2 – Výsledek simulace pro kmitočet f_1 | 63 |
| Obr. 4.3 – Blokové schéma filtru s kmitočtem f_2 | 65 |
| Obr. 4.4 – Výsledek simulace pro kmitočet f_2 | 65 |
| Obr. 4.5 – Blokové schéma filtru s kmitočtem f_3 | 67 |
| Obr. 4.6 – Výsledek simulace pro kmitočet f_3 | 67 |
| Obr. 4.7 – Výsledek simulace pro kmitočet f_4 | 69 |
| Obr. 4.8 – Blokové schéma filtru s kmitočtem f_4 | 69 |
| Obr. 4.9 – Vývojový diagram číslicového filtru..... | 72 |

SEZNAM TABULEK

| | |
|--|----|
| Tab. 1.1 – Koeficienty Butterworthovy aproximace (Brtník, Matoušek, 2014)..... | 35 |
| Tab. 1.2 – Vztahy pro přepočítání NDP na filtry (Brtník, Matoušek, 2014) | 36 |
| Tab. 2.1 – Parametry zdroje napájení (www.tme.eu, nedatováno)..... | 41 |
| Tab. 2.2 – Parametry A/Č převodníku (Matoušek, 2013) | 41 |
| Tab. 4.1 – Koeficienty pro filtr druhého řádu..... | 60 |
| Tab. 4.2 – Bilineární transformace dolní propusti na pásmovou propust..... | 61 |

ÚVOD

Selektivní voltmetry jsou speciálním případem pro měření napětí na několika konkrétních frekvencích, označované jako jednofrekvenční selektivní voltmetry. Dříve byly jejich filtry realizovány analogově, v dnešní době je možné selektivní voltmetry realizovat aplikací číslicových filtrů. Pomocí číslicového zpracování signálů dochází ke změně reprezentace signálu v čase spojitým na signál v čase diskrétním.

Motivací a důvodem pro tuto bakalářskou práci je fakt, že tyto selektivní měřiče úrovně tzv. jednofrekvenční (pevně naladěné) se užívaly v telekomunikacích. Hlavním cílem je číslicové signály analyzovat, modifikovat nebo z nich vybrat určitou informaci.

Tato práce se zabývá možností využití mikropočítače s architekturou AVR. Mikropočítač slouží jako číslicový filtr selektivního voltmetru. Oproti analogovým systémům má číslicové zpracování systémů několik výhod, jako např. flexibilita, reprodukovatelnost, spolehlivost.

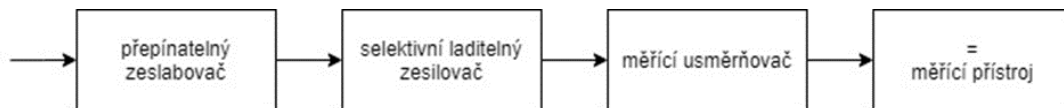
1 TEORETICKÁ ČÁST

1.1 SELEKTIVNÍ VOLTMETRY

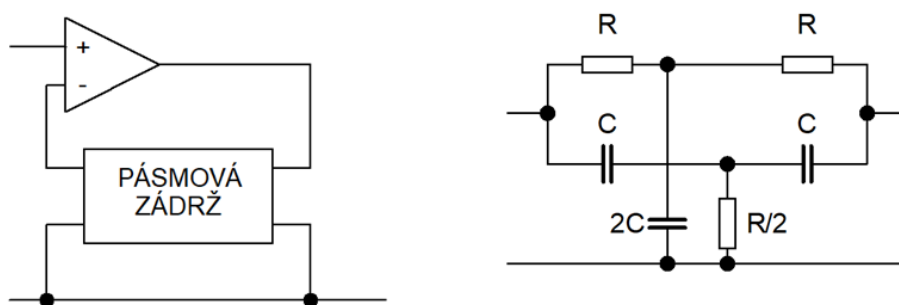
Pokud měříme velmi malá střídavá napětí, je nutné jejich značné zesílení před usměrněním. Zesílením se projeví vlastní šum zesilovače, který je úměrný šířce zesilovaného kmitočtového pásma. Zesilovač musí pracovat v úzkém kmitočtovém pásmu, aby šumové napětí bylo zanedbatelné oproti měřenému napětí. Výhodné je jejich použití i při měření určitého kmitočtu, které je obsaženo ve směsi jiných kmitočtů. Základem nízkofrekvenčních selektivních mikrovoltmetrů je laditelný úzkopásmový zesilovač. To je znázorněné blokovým schématem na obr. 1.1. Je tvořený úzkopásmovou zádrží ve smyčce záporné zpětné vazby a širokopásmovým zesilovačem. Na obr. 1.2 je zobrazen selektivní nízkofrekvenční zesilovač. Jeho kmitočtová charakteristika je stejná jako kmitočtová charakteristika jednoduchého rezonančního obvodu s činitelem jakosti

$$Q = \frac{1 + A_N}{4} \quad (1.1)$$

kde A_N – napěťové zesílení zesilovače.



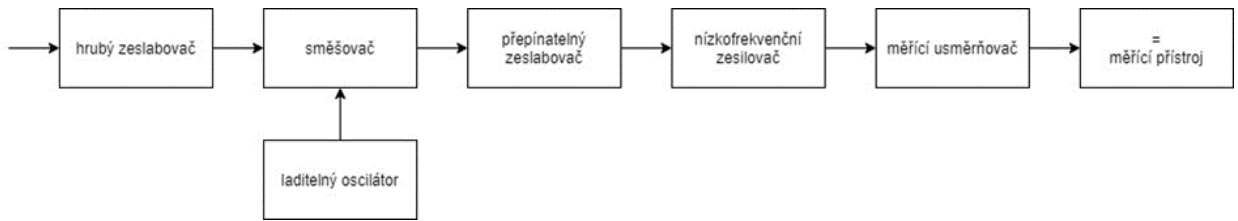
Obr. 1.1 – Blokové schéma nízkofrekvenčního selektivního voltmetru



Obr. 1.2 – Nízkofrekvenční selektivní zesilovač

a) základní zapojení, b) pásmová zádrž RC typu dvojitého T

Selektivní mikrovoltmetry pro vysoké a velmi vysoké kmitočty jsou většinou řešeny jako heterodynní, viz obr. 1.3. V nich se používá dvojího i trojího směšování, díky tomu se dosáhne dobré citlivosti, selektivity a stability. Ty jednodušší mají pouze jeden směšovač a oscilátor.



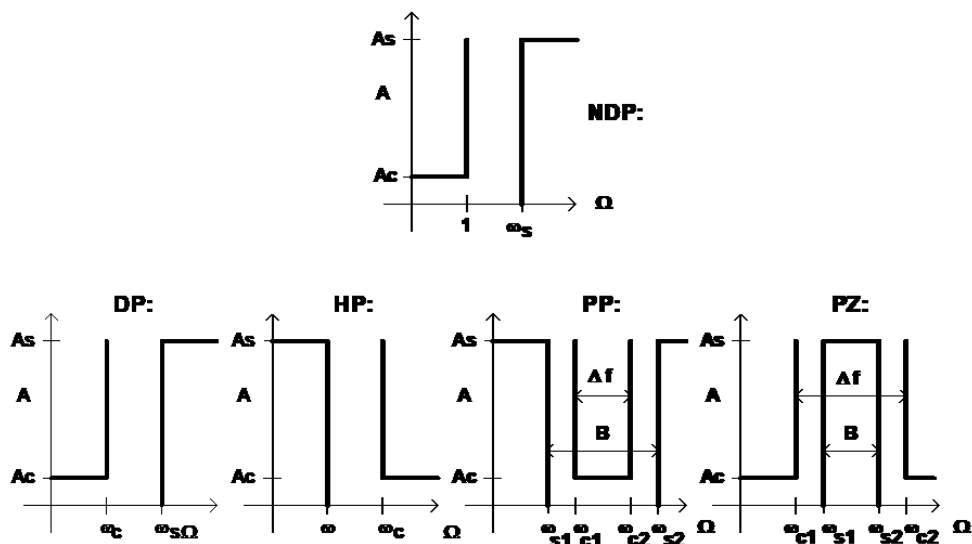
Obr. 1.3 – Blokové schéma heterodynního mikrovoltmetru s jedním směšováním

Ve směšovači dochází ke směšování střídavého napětí a napětí z oscilátoru. Pro zesílení složky v mezifrekvenčním zesilovači se nastaví kmitočet napětí oscilátoru tak, aby její kmitočet byl součtem (pro nízkofrekvenční mikrovoltmetry) nebo rozdílem (pro vysokofrekvenční mikrovoltmetry) kmitočtu napětí oscilátoru a kmitočtu měřeného napětí. Pak dochází k usměrnění a měření zesíleného napětí. Mezifrekvenční zesilovač určuje citlivost a selektivitu. Laděním oscilátoru se přeladuje mikrovoltmetr. Pomocí krystalového filtru v mezifrekvenčním zesilovači se docílí ostré selektivity.

Příkladem může být selektivní voltampérmetr AU 031 vyráběný někdejšími ZPA Trutnov pro měření systémů HDO, která se provádělo na frekvencích 1050 Hz, 750 Hz a 425 Hz. Takováto zařízení však lze v současnosti realizovat též aplikací filtrů číslicových, o čemž pojednává předložená práce.

1.2 ELEKTRICKÉ FILTRY

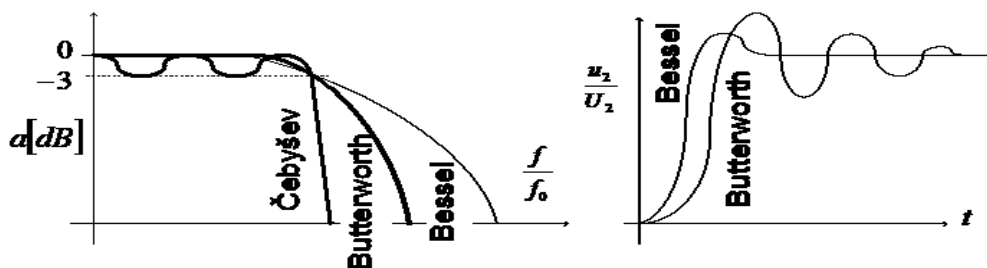
Elektrické filtry jsou obvody, které propouštějí signál s nulovým nebo nepatrným útlumem v určitém pásmu kmitočtů omezeném kmitočtem ω_c pro tzv. referenční, resp. normovanou dolní propust (NDP) podle obr.1.1. Jedná se o propustné pásmo. Mimo toto pásmo od kmitočtu ω_s signál nepropouští, prakticky pak s velkým útlumem. Mezi kmitočty ω_c a ω_s je pak tzv. přechodné pásmo. Filtry se dělí na čtyři základní typy: dolní propust (DP), horní propust (HP), pásmová propust (PP) a pásmová zadrž (PZ). Na obr. 1.4 je průběh útlumu A procházejících signálů u jednotlivých typů filtrů (Brtník, 2014).



Obr. 1.4 – Útlumové charakteristiky jednotlivých filtrů (Brtník, 2014)

Nejdůležitějším úkolem je odstranění nežádoucích a rušivých signálů, které nepatří do kmitočtového pásma užitečných signálů. Filtrací se spektrum signálu vhodně upravuje podle konkrétních požadavků. Pokud je modul přenosové charakteristiky konstantní, jedná se o tzv. fázovací článek, což je zvláštní případ filtru, kdy dochází jen ke změně fáze, ale nedochází ke změně amplitudy jednotlivých harmonických složek signálu (Hájek, Sedláček, 2002).

Při návrhu všech typů filtrů se vychází z normované dolní propusti (NDP). Účelem je zobecnění charakteristických vlastností filtrů. Použitelnost standardizovaných aproximací přenosové funkce pro všechny typy filtrů zajistíme transformací kmitočtového tolerančního schématu daného filtru na kmitočtové toleranční pole normované dolní propusti (Dostál, Vrba, 1993).



Obr. 1.5 – Přenosové charakteristiky filtrů a odezvy filtrů na jednotkový skok pro NDP (Brtník, 2014)

Je třeba zohlednit jak průběh odezvy v čase, tak průběh kmitočtové charakteristiky útlumu při návrhu filtru. Ta může být zadána jak tzv. analogovým prototypem filtru, tak i tolerančním polem útlumu (Brtník, 2014).

1.2.1 Znázornění činnosti analogového filtru typu dolní propust

U analogového signálu je nezávislou proměnnou čas, jedná se o reálnou funkci jedné reálné proměnné pro signál spojitý v čase (Davídek, Sovka, 1996).

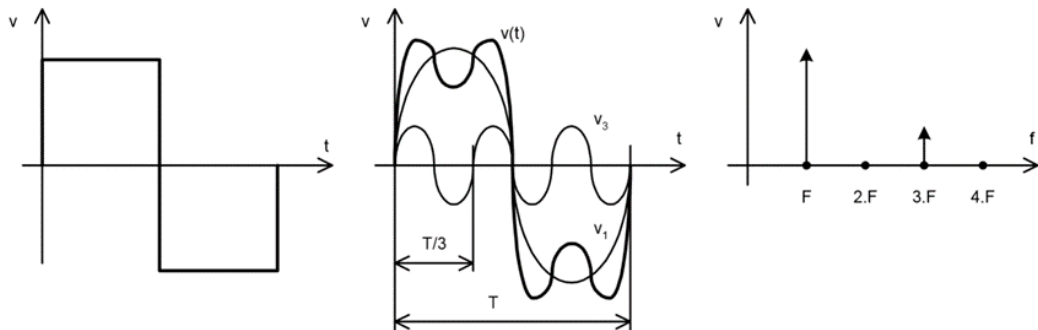
Signál přibližující se obdélníkovým impulsem lze aproximovat jako součet lichých harmonických, např. první v_1 a třetí v_3 harmonické podle vztahu:

$$v(t) = v_1 + v_3 = A_1 \sin \Omega t + A_3 \sin 3\Omega t \quad (1.2)$$

Spektrum takového signálu bude zahrnovat jednu spektrální čáru s amplitudou A_1 na kmitočtu Ω a druhou s amplitudou A_3 na kmitočtu 3Ω , jak je vidět z obr. 1.6 (Brtník, Matoušek, 2012).

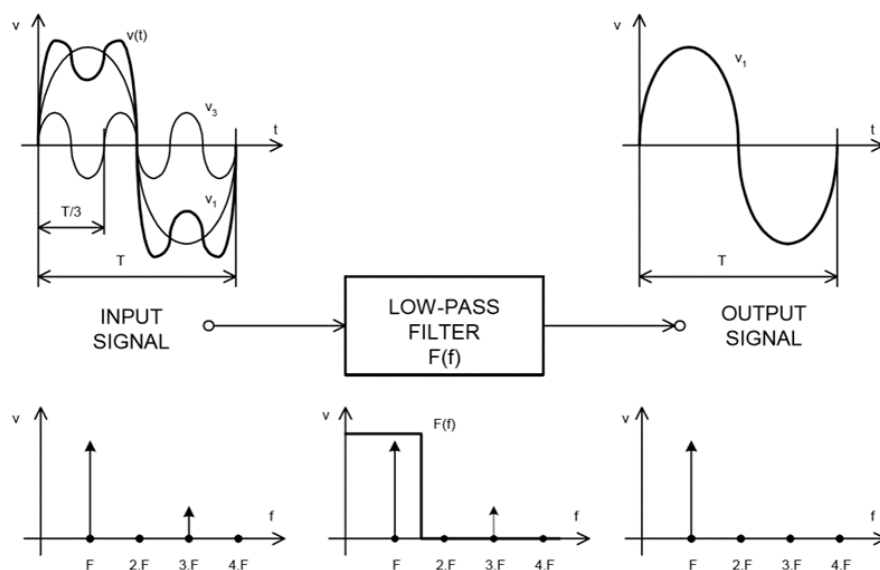
$$\Omega = 2\pi F \quad (1.3)$$

$$3\Omega = 3 \times 2\pi F \quad (1.4)$$



Obr. 1.6 – Téměř obdélníkové impulsy rozloženy na dva sinusové signály (Brtník, Matoušek, 2012)

U filtru typu dolní propust se potlačí složky s frekvencí větší, než je mezní kmitočet Ω_C a průchodem filtrem se tak upraví spektrum daného signálu. Časový průběh signálu na výstupu filtru je zcela odlišný od vstupního signálu, rozdíl je zřetelný z obr. 1.7. Zde na vstupu filtru je téměř obdélníkový průběh, konkrétně jeho základní a třetí harmonická, třetí harmonická je filtrem dolní propust potlačena, takže na výstupu filtru je pouze základní první harmonická, které odpovídá dle obrázku průběh sinusový (Brtník, Matoušek, 2012).

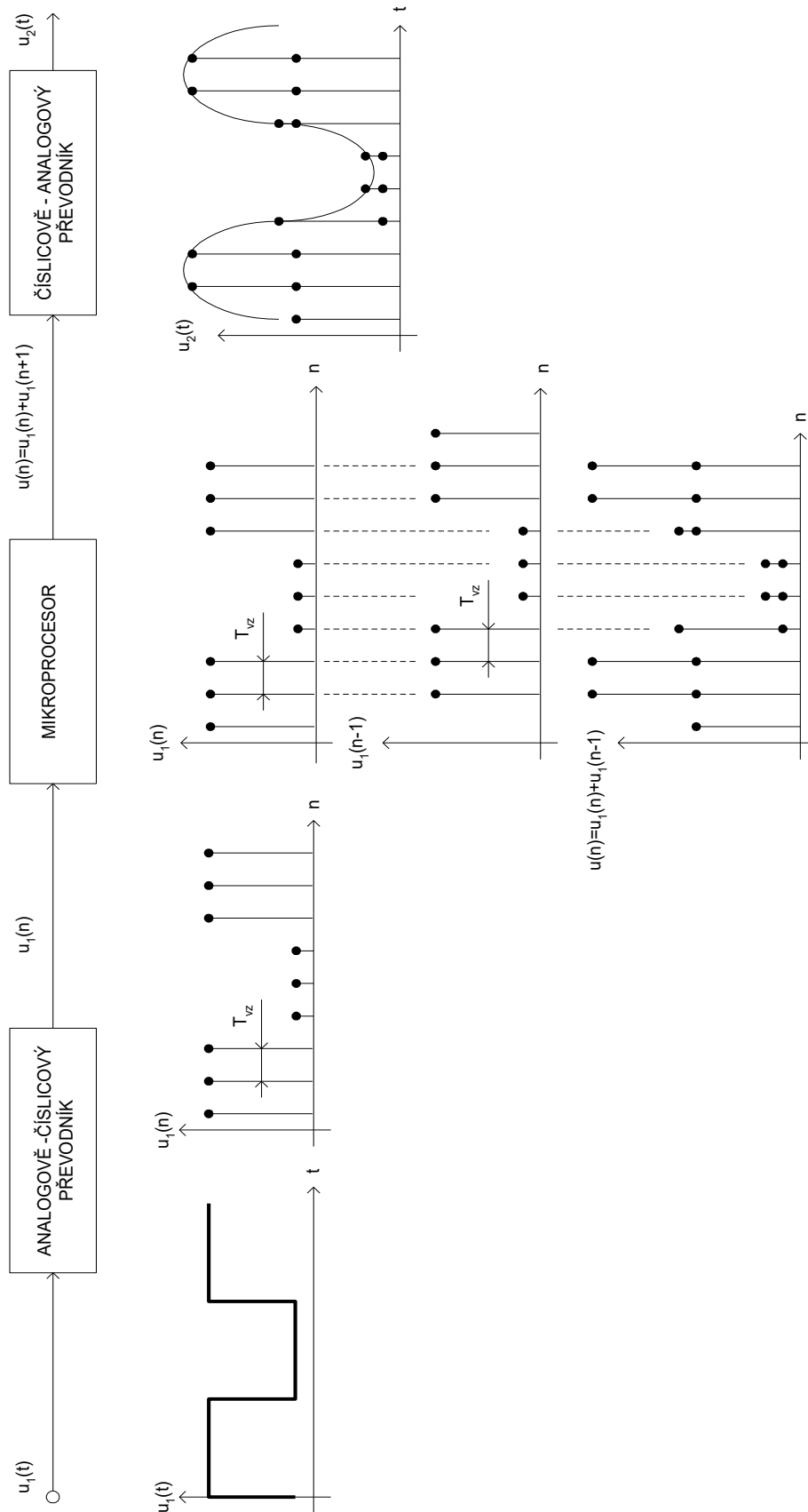


Obr. 1.7 – Časové průběhy a spektrum na vstupu a výstupu dolní propusti (Brtník, Matoušek, 2012)

1.2.2 Znázornění činnosti číslicového filtru typu dolní propust

Číslicovým signálem nazýváme signál diskrétní v čase a kvantovaný v úrovni. S tímto typem signálu pracují číslicové systémy (Davídek, Sovka, 1996).

Pokud budeme přičítat vzorky obdélníkového signálu, které jsou zpožděné o vzorkovací periodu, dosáhneme sinusového tvaru se stejnou periodou, stejně jako u analogového filtru typu dolní propust. Obecně lze uvést, že pouhým sčítáním a odčítáním aktuálních a předchozích vzorků signálu se dosáhne filtračních účinků. Princip číslicového filtru je znázorněn na obr. 1.8. Do analogově-číslcového převodníku a zároveň do vzorkovače vstupuje spojité napětí $u_1(t)$. Na jeho výstupu jsou pouze jeho vzorky $u_1(n)$ v diskrétních časových okamžicích vzdálených o vzorkovací periodu T_{VZ} , které dále postupují do mikroprocesoru. Zde se sčítá předchozí vzorek $u_1(n-1)$ a aktuální vzorek $u_1(n)$. Výsledek z mikroprocesoru $u(n)$ přechází do číslicově-analogového převodníku, kde se mění na analogový signál $u_2(n)$. Signál, který vznikne součtem aktuálního a předchozího vzorku odpovídá první harmonické vstupního napětí $u_1(t)$, takže opakovaným prováděním součtů vzorků získáme filtr typu dolní propust (Brtník, Matoušek, 2010).



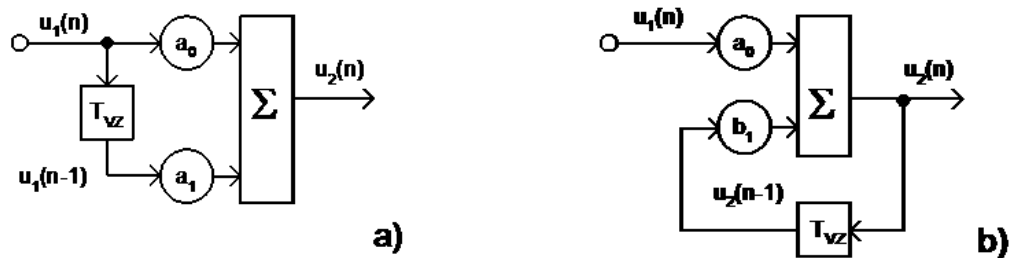
Obr. 1.8 – Názorný příklad činnosti číslicového filtru typu dolní propust (Brtník, Matoušek, 2010)

1.2.3 Popis činnosti filtrů

Blokové schéma na obr. 1.9 znázorňuje činnost filtru, kde probíhá součet aktuálního vzorku se vzorkem předchozím, který se získá zpožděním vzorku vstupního napětí o jednu vzorkovací periodu T_{VZ} . Pokud filtr nemá zpětnou vazbu, vstupní impuls zanikne po určité době, kdy projde zpožďovacími články. Jedná se o filtr s konečnou impulsní odezvou FIR. Filtr se zpětnou vazbou, kde může vstupní impuls kolovat teoreticky nekonečně dlouho, nazýváme filtr s nekonečnou impulsní odezvou IIR. (Brtník, Matoušek, 2012)

Schopnost filtrace závisí i na váze součtu aktuálního a zpožděného vzorku. Váhu znázorňují koeficienty a_i . Lze provádět i součet aktuálního vzorku se vzorkem zpožděným o vzorkovací periodu T_{VZ} výstupního signálu a analogicky vynásobeným koeficientem b_i . To lze popsat rovnicí (Brtník, Matoušek, 2010)

$$u_2(n) = a_0 u_1(n) + b_1 u_2(n - 1). \quad (1.5)$$



Obr. 1.9 – Blokové schéma filtru typu dolní propust (Brtník, Matoušek, 2010)

a) FIR, b) IIR

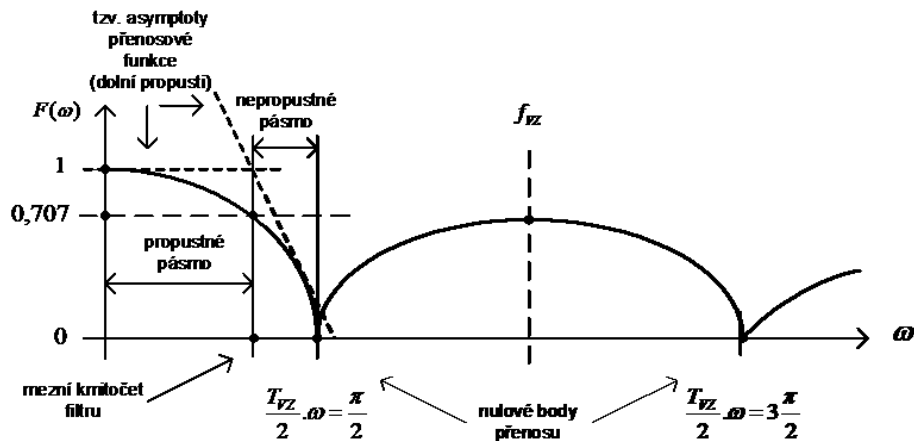
Pokud platí, že výstupní signál se rovná součtu aktuálního signálu a signálu zpožděného o vzorkovací periodu, lze ho popsat matematicky

$$u_2(t) = u_1(t) + u_1(t + T_{VZ}) = u_1(t) + u_1(t) e^{-j\omega T_{VZ}} = u_1(t) \times (1 + e^{-j\omega T_{VZ}}) = 2 u_1(t) e^{-j\omega \frac{T_{VZ}}{2}} \times \cos \omega \frac{T_{VZ}}{2} \quad (1.6)$$

v závislosti na proměnné frekvenci ω podle kosinové funkce $\cos \omega \times \frac{T_{VZ}}{2}$ se mění amplituda $u_2(t)$. U kosinového průběhu se maximální hodnota zmenšuje, což je dáno členem $e^{-j\omega T_{VZ}}$. Ten zajišťuje změnu znaménka kosinusové funkce ze záporného na kladné, výsledkem je absolutní hodnota této funkce. U přenosu $F(\omega)$ dochází ke stejné změně a platí pro něj

$$F(\omega) = \frac{u_2(t)}{u_1(t)} = 2 e^{-j\omega \frac{T_{VZ}}{2}} \times \cos \omega \frac{T_{VZ}}{2}. \quad (1.7)$$

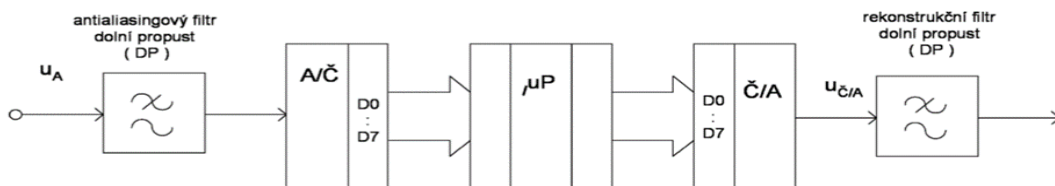
Filtr je realizován součtem a propouští frekvence od 0 do mezního kmitočtu neboli nepropustného pásma, závislost je zřetelná z obr. 1.10.(Brtník, Matoušek, 2014)



Obr. 1.10 – Graf přenosové funkce $F(\omega) = 2 \cdot e^{-j\omega \cdot T_{VZ}/2} \cdot \cos \omega \cdot \tau / 2$ dolní propusti (Brtník, Matoušek, 2014)

Na lichých násobcích $\pi / 2$ nastávají tzv. nulové body přenosu, kde se přenos rovná nule. Střídají se po π stejně jako póly přenosu, při nichž roste přenos nade všechny meze. Pro číslicové filtry je typická tato periodičita neboli aliasing (Brtník, Matoušek, 2014).

Pokud chceme zabránit aliasingu, což znamená falšování kmitočtu vzorkovaného analogového signálu, je potřeba dodržet Shannon-Kotelnikův vzorkovací teorém. Jedná se o vztah, díky kterému zabráníme ztrátě informace při digitalizaci analogového signálu. Vzorkovací frekvence A/Č převodníku musí být minimálně dvakrát větší než maximální frekvence ve spektru digitalizovaného signálu. Číslicové filtry využívají k omezení kmitočtového spektra vstupního signálu tzv. antialiasingový filtr. Ten je typu dolní propust, mezní kmitočet se rovná polovině vzorkovacího kmitočtu. Nad tímto mezním kmitočtem nesmí být v ideálním případě přenášén žádný signál. Pro zpětný převod z navzorkovaného signálu na původní analogový signál se používá rekonstrukční filtr a musí být na výstupu zařízení, které číslicově zpracovává analogový signál, viz obr. 1.11. Filtr odstraní všechny složky o kmitočtech vyšších, než je polovina vzorkovacího kmitočtu ze spektra navzorkovaného signálu, je typu dolní propust (Hájek, Sedláček, 2002).



Obr. 1.11 – Vstupní antialiasingový a výstupní rekonstrukční analogový filtr u číslicového filtru (Brtník, Matoušek, 2010)

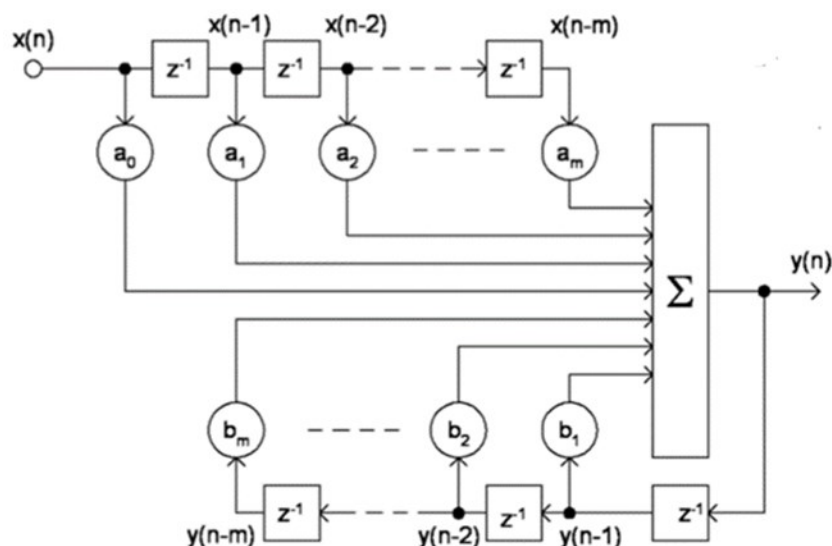
U sinové závislosti bude výstupní signál výsledkem aktuálního signálu a zpožděného signálu o vzorkovací frekvenci

$$\begin{aligned}
 u_2(t) &= u_1(t) - u_1(t + T_{VZ}) = u_1(t) - u_1(t) e^{-j\omega T_{VZ}} = \\
 &u_1(t) (1 - e^{-j\omega T_{VZ}}) = 2 u_1(t) e^{-j\omega \frac{T_{VZ}}{2}} \times \sin\omega \frac{T_{VZ}}{2},
 \end{aligned}
 \tag{1.8}$$

podle sinové funkce ω se mění amplituda $u_2(t)$ a z toho vyplývá, že rozdílem se realizuje horní propust. Při kaskádním spojení dolní a horní propusti může vzniknout pásmová propust, pokud bude dodrženo, že mezní kmitočet dolní propusti bude vyšší než mezní kmitočet horní propusti ($f_{HP} \ll f_{DP}$). Kmitočty větší, než je f_{HP} projdou horní propustí, ale kmitočty větší, než f_{DP} neprojdou propustí dolní. Pásmová propust bude dána analogickou kombinací součtů a rozdílů, některé koeficienty budou s kladným, jiné se záporným znaménkem, protože dolní propust je dána rozdílem vzorků a horní propust součtem vzorků (Brtník, Matoušek, 2014).

1.2.4 Obecná rovnice pro realizaci číslicového filtru

IIR filtr může mít více než jeden zpožďovací článek a může provádět součet více předchozích vzorků. Výstupní signál se rovná součtu aktuálního vstupního vzorku $x(n)$ vynásobeným koeficienty a_m s předchozími vstupními vzorky $x(n-m)$ se zpožděnými vzorky výstupního signálu $y(n-m)$ násobenými koeficienty b_m (Brtník, Matoušek, 2010).



Obr. 1.12 – Blokové schéma filtru IIR o m -článcích (Brtník, Matoušek, 2010)

Diferenční rovnice podle obr. 1.12 má tvar

$$y(n) = a_0 x(n) + a_1 x(n-1) + a_2 x(n-2) + \dots + a_m x(n-m) + b_1 y(n-1) + \dots + b_m y(n-m). \quad (1.9)$$

Zavedením operátoru z ve tvaru z^{-1} lze rovnici upravit

$$y(z) = a_0 x(z) + a_1 x(z) z^{-1} + a_2 x(z) z^{-2} + \dots + a_m x(z) z^{-m} + b_1 y(z) z^{-1} + \dots + b_m y(z) z^{-m}. \quad (1.10)$$

Pro vyjádření přenosové funkce $F(z)$ číslicového filtru je potřeba upravit $x(z)$, $y(z)$ a vyjádřit jejich poměr

$$x(z) \times (a_0 + a_1 z^{-1} + \dots + a_m z^{-m}) = y(z) \times (1 - b_1 z^{-1} - \dots - b_m z^{-m}) \quad (1.11)$$

$$F(z) = \frac{y(z)}{x(z)} = \left(\frac{a_0 + a_1 z^{-1} + \dots + a_m z^{-m}}{1 - b_1 z^{-1} - \dots - b_m z^{-m}} \right) \quad (1.12)$$

$$F(z) = - \left(\frac{a_m z^{-m} + \dots + a_1 z^{-1} + a_0}{b_m z^{-m} + \dots + b_1 z^{-1} - 1} \right). \quad (1.13)$$

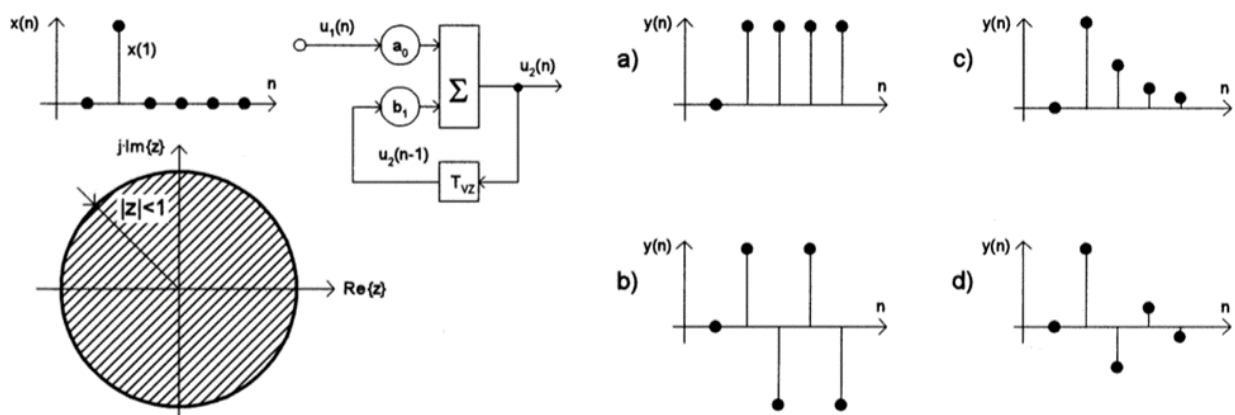
Pro návrh filtru lze využít metodu porovnání přenosové funkce $F(z)$ a přenos $F(s)$, který má obecnou rovnici ve tvaru

$$F(s) = \left(\frac{a_m s^m + \dots + a_1 s + a_0}{b_n s^n + \dots + b_1 s + b_0} \right) \quad (1.14)$$

1.2.5 Stabilita číslicového filtru

Stabilita číslicového filtru je dána jeho výstupní odezvou, která musí být konečná pro každý konečný vstupní signál. Nastavením pólů v přenosové rovnici daného filtru zajistíme tuto podmínku stability. Pokud leží všechny póly v jednotkové kružnici, jedná se o stabilní číslicový filtr. V opačném případě, kdy jsou póly mimo jednotkovou kružnici, jedná se o nestabilní číslicový filtr. Pro nestabilitu systému stačí, když bude alespoň jeden pól mimo jednotkovou kružnici. Přechod mezi stabilní a nestabilní oblastí tvoří jednotková kružnice a tento přechod se nazývá mez stability. Pokud leží pól nebo póly na mezi stability, musí se celková stabilita systému zjistit pomocí početních metod (Davídek, Sovka, 1996).

Pro stabilitu IIR filtru se musí vzorek po každém průchodu zpětnovazební smyčkou zmenšit. Dodržením této podmínky se bude vzorek zmenšovat až na nulu s rostoucím časem. Pokud se vzorek nebude zmenšovat, tak by teoreticky jeden vzorek mohl cyklicky putovat nekonečně dlouho ve zpětnovazební smyčce. Z filtru by se pak stal vlastně oscilátor. Pro hodnoty koeficientů $a_0 = 1$, $b_1 = 1$ budou impulsy $y(n)$ stejně velké a bude jich teoreticky nekonečně mnoho, jak je vidět z obr. 1.13 průběhu a). Pro hodnoty koeficientů $a_0 = 1$, $b_1 = -1$ se bude posloupnost výstupních impulsů $y(n)$ střídát podle znaménka, ale jejich velikost bude stejná viz průběh b) na obr. 1.13. Stejně jako v předchozím případě se obvod v čase jdoucím k nekonečnu neustálí a filtr je nestabilní. Při zmenšení koeficientu ve zpětné vazbě například na $b_1 = 0,5$ a $a_0 = 1$ se začnou impulsy $y(n)$ zmenšovat až k nule v případě, že jich bude nekonečný počet. Na obr. 1.13 je to znázorněno průběhem c). V případě změny znaménka u koeficientu $b_1 = -0,5$ a $a_0 = 1$, se budou impulsy $y(n)$ zmenšovat až k nule, stejně jako u předchozího průběhu, pokud jich bude nekonečný počet. Jediný rozdíl bude ve střídání znaménka, jak je graficky znázorněno průběhem d) na obr. 1.13. IIR filtr bude tedy stabilní, bude-li koeficient $|b_1| < 1$. (Brtník, Matoušek, 2010)



Obr. 1.13 – K otázce stability filtru IIR (Brtník, Matoušek, 2010)

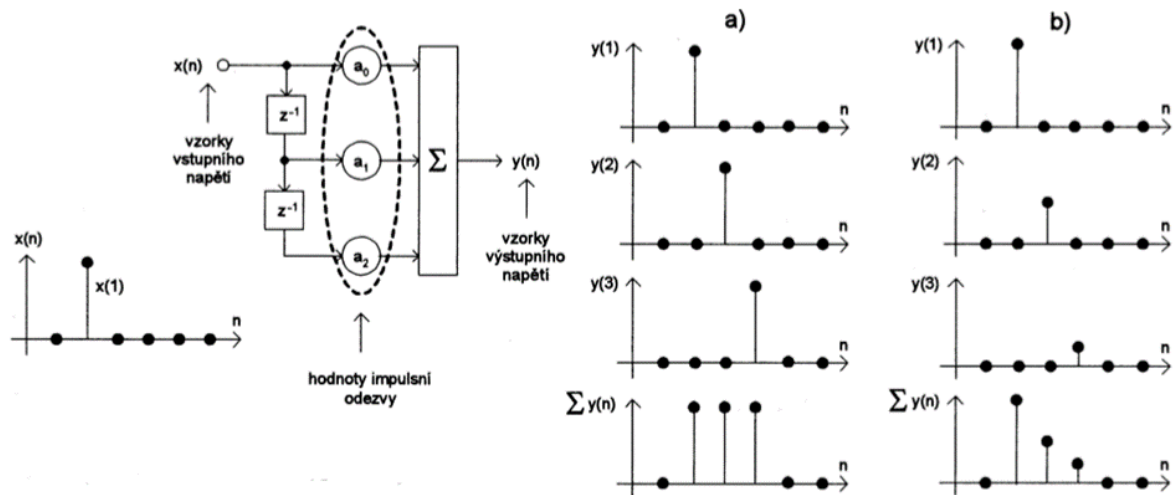
U filtru FIR vstupní impuls projde zpožďovacími členy a zanikne vždy, nezáleží na velikosti koeficientu a_i . Např. pro filtr popsany rovnicí

$$y(n) = a_0 x(n) + a_1 x(n - 1) + a_2 x(n - 2) \quad (1.15)$$

jsou hodnoty odezvy $y(n)$ na vstupní signál $x(1)$ pro koeficienty $a_0 = a_1 = a_2 = 1$ graficky znázorněny jako průběh a) na obr. 1.14. Snížení hodnot koeficientů je znázorněno průběhem b) na obr. 1.14, velikost výstupních impulsů $y(n)$ jde k nule. Koeficienty a_i udávají průběh odezvy filtru na jednotkový impuls svojí hodnotou. Bez ohledu na koeficienty a_i je filtr typu FIR vždy stabilní, protože po úpravě rovnice:

$$F(z) = (a_0 + a_1 z^{-1} + a_2 z^{-2}) \times \frac{z^2}{z^2} = \frac{(a_0 z^2 + a_1 z^1 + a_2)}{z^2} \quad (1.16)$$

zjistíme, že ve jmenovateli je jediný kořen $z^2 = 0$, leží teda u počátku souřadnic jednotkové kružnice. (Brtník, Matoušek, 2010)



Obr. 1.14 – K otázce stability filtru FIR (Brtník, Matoušek, 2010)

1.2.6 Závislost frekvenční charakteristiky na poloze pólů

Frekvenční charakteristika přenosu $F(\omega)$ filtru prvního řádu se dá popsat vztahem

$$F(s) = \frac{K}{s - p} \quad (1.17)$$

kde p – pól přenosu filtru $F(\omega)$

s – symbolický kmitočet

Pokud se změní kmitočet o Δp , tak bude mít frekvenční charakteristika tvar

$$F(s) = \frac{K}{s - (p + \Delta p)}. \quad (1.18)$$

Frekvenční charakteristika druhého řádu se dá popsat vztahem

$$F(s) = \frac{K}{(s - p)^2} = \frac{K}{s^2 - 2sp + p^2}. \quad (1.19)$$

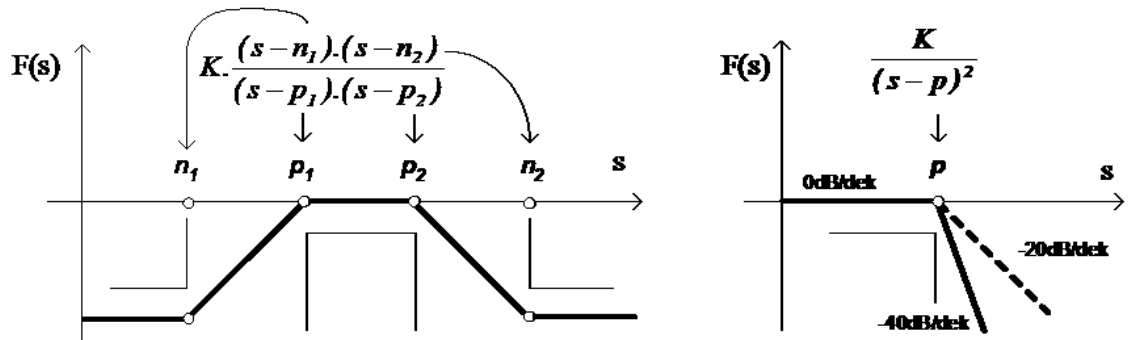
Při změně kmitočtu o Δp , dojde k úpravě na

$$F(s) = \frac{K}{[s - (p + \Delta p)]^2} = \frac{K}{s^2 - 2s p + 2s \Delta p + p^2 + 2p \Delta p + \Delta p^2}. \quad (1.20)$$

Porovnáním jmenovatelů zlomků prvního a druhého řádu zjistíme, že filtr druhého řádu má jmenovatel rozšířený o člen $2s + 2p + \Delta p$, kterým je násobeno Δp . To způsobuje odchylku průběhu $F(\omega)$, která je větší než odchylka filtru prvního řádu. Tímto se dá zobecnit, že citlivost kmitočtové charakteristiky filtru bude vyšší s vyšším řádem filtru (Brtník, Matoušek, 2014).

Filtr typu pásmová propust se skládá z filtru typu dolní propust, který realizuje zpožděný vzorek s kladným znaménkem a filtru typu horní propust, ten realizuje zpožděný vzorek se

záporným znaménkem. Nejnižším používaným řádem pro $F(\omega)$ je tedy druhý řád, tzv. bikvad. Při realizaci filtrů vyšších řádů se upřednostňuje jejich složení z bikvadů, kvůli výše uvedeným důvodům. Změny asymptoty charakteristiky $F(\omega)$ vyvolá každý nulový bod o $+20\text{dB/dek}$, zatímco každý pól -20dB/dek ve frekvenční charakteristice filtru (Brtník, Matoušek, 2014).



Obr. 1.15 – Vliv nulových bodů a pólů na průběh kmitočtové charakteristiky $F(s)$ (Brtník, Matoušek, 2014)

1.2.7 Impulsní charakteristika číslicového filtru

Impulsní charakteristika $h(t)$ je odezva obvodu na jednotkový (neboli Diracův) impuls $\delta(t)$. Ten je derivací jednotkového skoku $1(t)$. Odezva na jednotkový skok je přechodová charakteristika $g(t)$. Vyjádření vztahů

$$\begin{aligned} 1(t) &= 0 \text{ pro } t < 0 \\ 1(t) &= 1 \text{ pro } t > 0 \end{aligned} \quad (1.21)$$

$$\delta(t) = \frac{d1(t)}{dt} \quad (1.22)$$

$$h(t) = \frac{dg(t)}{dt}. \quad (1.23)$$

Přenos obvodu $F(p)$ bude dán poměrem Laplaceových obrazů jednotkového skoku a přechodové charakteristiky

$$L\{1(t)\} = \frac{1}{p} = U_1(p) \quad (1.24)$$

$$L\{g(t)\} = G(p) = U_2(p) \quad (1.25)$$

$$F(p) = \frac{U_2(p)}{U_1(p)} = G(p) p. \quad (1.26)$$

Laplaceovou transformací impulsní charakteristiky $h(t)$ vyjde vztah

$$h(t) = L^{-1}\{p G(p)\} = L^{-1}\{F(p)\} \quad (1.27)$$

$$F(p) = L\{h(t)\} \quad (1.28)$$

a přenos bude analogicky

$$F(z) = Z\{h(n)\}. \quad (1.29)$$

Z-transformací přenosové funkce bude

$$u_2(n) = a_0 u_1(n) + a_1 u_1(n - 1) \quad (1.30)$$

$$F(z) = a_0 + a_1 z^{-1}, \quad (1.31)$$

obecně

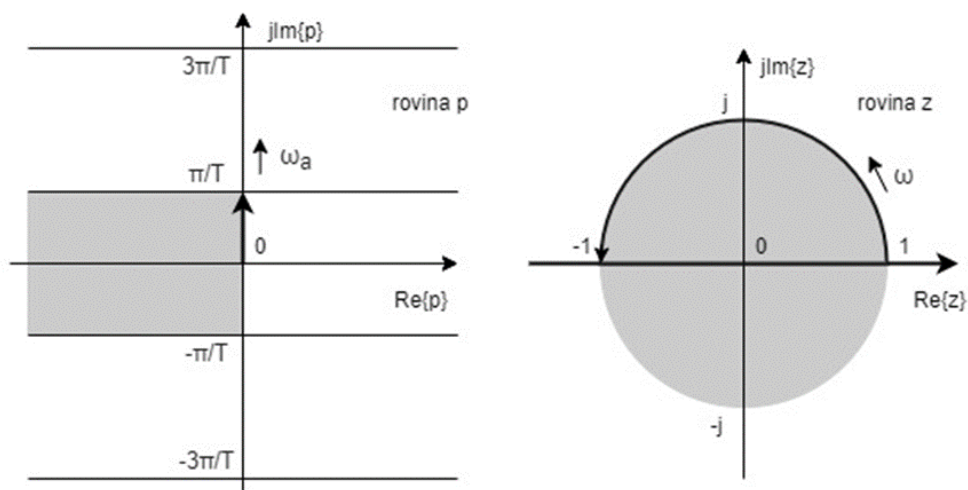
$$F(z) = a_0 + a_1 z^{-1} + \dots + a_N z^{-N}. \quad (1.32)$$

Jednotlivé členy konvergují k nule, a to i v případě, že bude z záporné, pokud $|z| < 1$. Pro $z = 1$ budou jednotlivé členy stejné. Při $z > 1$ budou jednotlivé členy růst, takže obvod se rozkmitává (Brtník, Matoušek, 2010).

1.3 REALIZACE ČÍSLICOVÝCH FILTRŮ

Tato kapitola popisuje vztah mezi analogovými a číslicovými systémy. Nejčastěji jde o nahrazení číslicových filtrů pomocí analogových prototypů. Přesná náhrada není možná, ale vždy se jedná o náhradu, která splňuje předem dané podmínky. Na příkladech budou zobrazeny postupy návrhu analogového prototypu metodou s využitím bilineární transformace a impulsní invariance.

Pro zachování frekvenčních vlastností transformovaného systému musí být dodržena podmínka, kdy na jednotkovou kružnici z -roviny by měla být zobrazena imaginární osa $j\Omega$ p -roviny. To znamená, že pokud je původní analogový filtr typu dolní propust, musí být i číslicový filtr typu dolní propust a splňovat toleranční schéma. Zachování stability zajistí póly přenosové funkce, které má stabilní analogový filtr v levé části p -roviny. Póly číslicového filtru po transformaci do z -roviny musí ležet uvnitř jednotkové kružnice pro splnění podmínky. Zobrazení p -roviny na z -rovinu je na obr. 1.16 (Smékal, 1993).



Obr. 1.16 – Transformace roviny p na rovinu z (Smékal, 1993)

1.3.1 Metoda rozmístění nulových bodů a pólů

Návrh číslicového filtru je nejjednodušší pomocí rozmístování nulových bodů a pólů přenosové funkce $F(z)$. Ta se dá vyjádřit pomocí podílu polynomů čitatele a jmenovatele

$$F(z) = \frac{a_0 + a_1 z^{-1} + \dots + a_M z^{-M}}{1 + b_1 z^{-1} + \dots + b_N z^{-N}}. \quad (1.33)$$

Mocniny z lze snadno převést ze záporných na kladné takto

$$\begin{aligned} \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}} &= \frac{(a_0 + a_1 z^{-1} + a_2 z^{-2})}{(1 + b_1 z^{-1} + b_2 z^{-2})} \times \frac{z^2}{z^2} = \\ &= \frac{a_0 z^2 + a_1 z^{-1} z^2 + a_2 z^{-2} z^2}{z^2 + b_1 z^{-1} z^2 + b_2 z^{-2} z^2} = \frac{a_0 z^2 + a_1 z^1 + a_2}{z^2 + b_1 z^1 + b_2}, \end{aligned} \quad (1.34)$$

a součin se dá upravit do tvaru

$$(p - a) \times (p - b) = p^2 + p \times (-a - b) + a b. \quad (1.35)$$

Tvar funkce lze přepsat do následující podoby

$$F(z) = \frac{a_0 \times (z - n_1) \times (z - n_2) \times \dots \times (z - n_M)}{(z - p_1) \times (z - p_2) \times \dots \times (z - p_N)}. \quad (1.36)$$

Pokud je potřeba potlačit průchod nějakého kmitočtu, položí se čítec nul, tím se stane i přenos kmitočtu nulový a získá se koeficient n_k . Koeficienty n_k jsou kořeny čitatele a tzv. nulové body. Naopak pro zdůraznění kmitočtu se položí nule jmenovatel přenosové funkce. Kořeny jmenovatele jsou tzv. póly p_i , při nichž je převrácená hodnota přenosové funkce rovna nule (Brtník, Matoušek, 2010).

1.3.2 Metoda impulsní invariance

Impulsní invariancí se zajistí, že číslicový filtr bude mít stejný časový průběh ustálení přechodového děje při buzení impulsním signálem jako jeho analogový prototyp. Ustálení bez překmitů se děje po exponenciální funkci

$$u = U e^{\delta \times t}, \quad (1.37)$$

obecně s překmity

$$u = (U e^{j \times \omega \times t}) e^{\delta \times t} = U e^{a \times t}. \quad (1.38)$$

Laplaceovým obrazem časového průběhu je frekvenční závislost, to znamená, že pro přechodový děj lze užít Laplaceovu transformaci a platí pro něj:

$$L\{U e^{a \times t}\} = U \frac{1}{s - a}. \quad (1.39)$$

V časové oblasti pro $F(s)$ musí mít póly přenosové funkce stejnou polohu jako v oblasti proměnné z , protože obě oblasti obsahují proměnnou s . Oblast proměnné z pro $F(z)$ je vyjádřena vztahem (Brtník, Matoušek, 2014)

$$z = e^{j \times \omega \times T_V z} \quad (1.40)$$

Pro získání popisu v oblasti operátoru z se provede z -transformace součtu parciálních zlomků, které vzniknou rozkladem kmitočtové závislosti přenosu filtru. Z tohoto důvodu se upraví výraz pro $F(s)$ do tvaru:

$$F(s) = \frac{a_M s^M + \dots + a_1 s + a_0}{b_N s^N + \dots + b_1 s + b_0} = \frac{a_M s^M + \dots + a_1 s + a_0}{b_N (s - p_N) \dots (s - p_1)} = \sum_{n=1}^{n=N} \frac{A_n}{s - p_n}, \quad (1.41)$$

kde p_n, \dots, p_l – póly $F(s)$.

Zpětnou Laplaceovou transformací dostaneme

$$L^{-1} \left\{ \sum \frac{A_n}{s - p_n} \right\} = \sum A_n e^{p_n \times t}. \quad (1.42)$$

Pomocí z -transformace se zavede operátor

$$z = e^{j \times \omega \times T_V z} \quad (1.43)$$

$$Z \left\{ \sum A_n e^{p_n \times t} \right\} = \sum A_n \frac{z}{z - e^{p_n \times t}} \times \frac{z^{-1}}{z^{-1}} = \sum A_n \times \frac{1}{1 - z^{-1} e^{p_n \times t}}. \quad (1.44)$$

Pro maximální kmitočet signálu ω opakujících se impulsů, musí být dodržen Shanonnův vzorkovací teorém

$$\omega = \frac{\omega_{VZ}}{2} = \frac{2\pi f_{VZ}}{2} = \pi f_{VZ}, \quad (1.45)$$

kmitočet ω vztažený ke vzorkovacímu kmitočtu v oblasti z je reprezentovaný hodnotou π , platí tedy vztah

$$\pi = \frac{\omega}{f_{VZ}} = \omega T_{VZ}, \quad (1.46)$$

vztah pro $F(z)$ je teda ještě potřeba vynásobit vzorkovací periodou T_{VZ}

$$F(z) = T_{VZ} \times \frac{A_n}{1 - z^{-1} e^{p_n \times t}}. \quad (1.47)$$

Rovnice se rozšíří pomocnou proměnnou $P(z)$, kde se z části $P(z)/X(z)$ vyjádří $P(z)$ a z druhé části $Y(z)/P(z)$ se vyjádří $Y(z)$

$$F(z) = \frac{Y(z)}{X(z)} = \frac{Y(z)}{X(z)} \times \frac{P(z)}{P(z)} = \frac{Y(z)}{P(z)} \times \frac{P(z)}{X(z)}. \quad (1.48)$$

Z tohoto vztahu lze vytvořit blokové schéma filtru, z něj se sestaví vývojový diagram, který se dá vytvořit program přepsáním do programovacího jazyka pro použitý procesor (Brtník, Matoušek, 2014).

1.3.3 Metoda bilineární transformace

Bilineární transformace je další způsob převedení z oblasti symbolického kmitočtu s do oblasti operátoru z . Díky této transformaci se dosáhne maximální možné shody průběhu charakteristiky navrhovaného číslicového filtru $F(z)$ a frekvenční charakteristiky analogového prototypu filtru $F(s)$ (Brtník, Matoušek, 2014).

Rovnice analogového filtru je důležitá při návrhu číslicového filtru. Jde o přenos $F(s)$ v oblasti $s = j\omega$

$$F(s) = \frac{U_2(s)}{U_1(s)}. \quad (1.49)$$

Následujícím vztahem při přechodu z $F(s)$ do oblasti se substitucí změny kmitočty

$$s = \frac{2}{T_{VZ}} \times \frac{1-z^{-1}}{1+z^{-1}}. \quad (1.50)$$

Změně kmitočtů je možné zabránit použitím vztahu:

$$\tilde{\Omega} = \frac{2}{T_{VZ}} \times \operatorname{tg} \frac{\Omega \times T_{VZ}}{2}. \quad (1.51)$$

Metoda vychází ze známého vzorkovacího kmitočtu f_{VZ} , ze známých mezních kmitočtů propustného pásma f_c , zadané aproximační funkce a zvlnění útlumu v propustném pásmu A_c , kmitočtu f_s , při kterém má mít filtr útlum aspoň A_s (Brtník, Matoušek, 2014).

Dosažením do následujících vztahů se předzkreslí kmitočty a převedou se na Ω_C a Ω_S

$$\Omega_C = \frac{2}{T_{VZ}} \times \operatorname{tg} \frac{2\pi f_c T_{VZ}}{2} \quad (1.52)$$

$$\Omega_S = \frac{2}{T_{VZ}} \times \operatorname{tg} \frac{2\pi f_s T_{VZ}}{2}. \quad (1.53)$$

Pro stanovení řádu filtru Butterworthovou aproximací je třeba znát činitel selektivity filtru a útlumový činitel filtru. Činitel selektivity se stanoví podílem předzkreslených kmitočtů:

$$k = \frac{\Omega_S}{\Omega_C} \quad (1.54)$$

a útlumový činitel filtru

$$d = \frac{10^{0,1 \times A_S} - 1}{10^{0,1 \times A_C} - 1}. \quad (1.55)$$

Tab. 1.1 – Koeficienty Butterworthovy aproximace (Brtník, Matoušek, 2014)

| n | a ₀ | a ₁ | a ₂ | a ₃ | a ₄ | a ₅ | a ₆ | a ₇ | a ₈ | a ₉ |
|---|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 1 | 1 | 1 | | | | | | | | |
| 2 | 1 | 1,4142 | 1 | | | | | | | |
| 3 | 1 | 2 | 2 | 1 | | | | | | |
| 4 | 1 | 2,6131 | 3,4121 | 2,6121 | 1 | | | | | |
| 5 | 1 | 3,2360 | 5,2360 | 5,2360 | 3,2360 | 1 | | | | |
| 6 | 1 | 3,8637 | 7,4641 | 9,1416 | 7,4641 | 3,8637 | 1 | | | |
| 7 | 1 | 4,4939 | 10,097 | 14,591 | 14,591 | 10,097 | 4,4939 | 1 | | |
| 8 | 1 | 5,1258 | 13,137 | 21,846 | 25,688 | 21,846 | 13,137 | 5,1258 | 1 | |
| 9 | 1 | 5,7587 | 16,581 | 31,163 | 41,986 | 41,986 | 31,163 | 16,581 | 5,7587 | 1 |

Řád filtru se stanoví vztahem

$$n \geq \frac{\log d}{2 \times \log k}. \quad (1.56)$$

Přenosová funkce filtru $F(s)$ v oblasti s se sestaví pomocí hodnot a_i , které se vyberou z tab. 1.1 koeficientů přenosové funkce pro příslušnou aproximaci podle řádu filtru. Polynom přenosové funkce $F(s)$

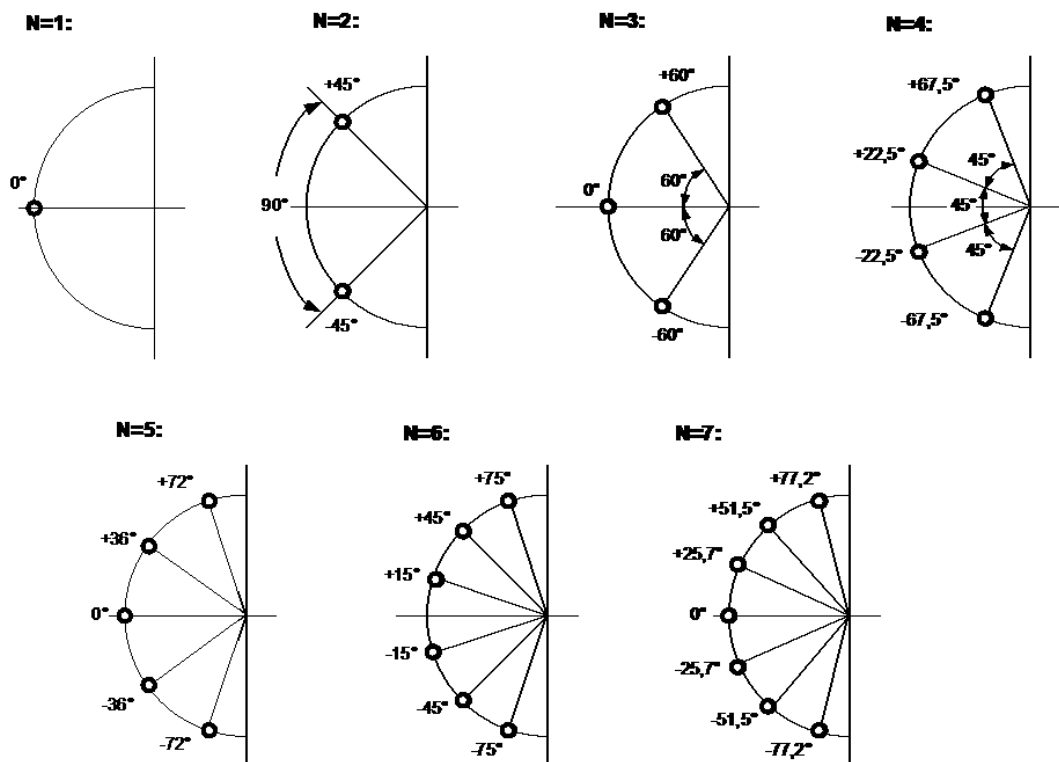
$$F(s) = \frac{1}{\sum_{i=0}^N a_i \times s^i} \quad (1.57)$$

Vztahem podle typu filtru, které jsou uvedeny v tab. 1.2 se pomocí bilineární transformace přejde z oblasti $s = j\omega$ do oblasti $z = e^{j\omega T_{VZ}}$. Přepočítání na konkrétní kmitočty se děje členem Ω .

Tab. 1.2 – Vztahy pro přepočítání NDP na filtry (Brtník, Matoušek, 2014)

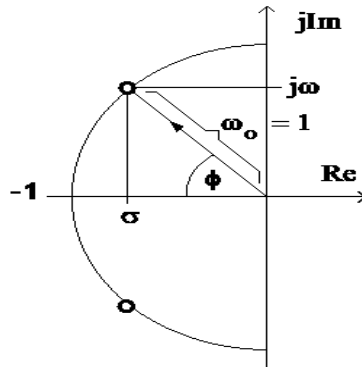
| Dolní propust | Horní propust | Pásmová propust | Pásmová zadrž |
|---|--|---|---|
| $s \rightarrow \frac{2}{\Omega T_{VZ}} \frac{z-1}{z+1}$ | $s \rightarrow \frac{2\Omega}{T_{VZ}} \frac{z-1}{z+1}$ | $s \rightarrow \beta \frac{z^2 - 2\alpha z + 1}{z^2 - 1}$ | $s \rightarrow \frac{1}{\beta} \frac{z^2 - 1}{z^2 - 2\alpha z + 1}$ |
| | | $\alpha = \frac{4 - \Omega^2 T^2}{4 + \Omega^2 T^2}$ | $\alpha = \frac{4 - \Omega^2 T^2}{4 + \Omega^2 T^2}$ |
| | | $\beta = \frac{4 + \Omega^2 T}{2\Delta\Omega T}$ | $\beta = \frac{4 + \Omega^2 T}{2\Delta\Omega T}$ |

Póly Butterworthových polynomů jsou pravidelně rozmístěny na půlkružnici, viz obr. 1.17 a z nich vycházejí hodnoty těchto polynomů.



Obr. 1.17 – Rozložení pólů Butterworthových polynomů na kružnici pro různé řády (Brtník, Matoušek, 2014)

Podle obr. 1.17 se zjistí, kde leží póly na jednotkové kružnici pro určitý řád filtru. Pro řád $N = 1$ je pól v poloze $s = -1$, Butterworthův polynom bude dán vztahem $s+1$. Pro druhý řád $N = 2$ jsou póly symetricky rozmístěny kolem vodorovné osy v polohách $\pm \cos 45^\circ$. Butterworthův polynom druhého řádu je na obr. 1.18.



Obr. 1.18 – K Butterworthovu polynomu druhého řádu (Brtník, Matoušek, 2014)

Souřadnice pólu jsou tedy

$$\begin{aligned} -\cos 45^\circ + j \sin 45^\circ &= -\sigma + j\omega = s \\ -\cos 45^\circ - j \sin 45^\circ &= -\sigma - j\omega = s. \end{aligned} \quad (1.58)$$

Ze souřadnic se vytvoří polynom

$$(s + \sigma + j\omega) \times (s + \sigma - j\omega) = s^2 + 2s\sigma + \sigma^2 + \omega^2. \quad (1.59)$$

Podle obr. 1.18 je využita Pythagorova věty a pro jednotkovou kružnici platí

$$\sigma^2 + \omega^2 = 1, \quad (1.60)$$

pro reálnou složku platí

$$\sigma = \cos \phi. \quad (1.61)$$

Z toho plyne polynom

$$\begin{aligned} s^2 + 2s \cos \phi + 1 &= s^2 + 2s \cos 45^\circ + 1 = s^2 + 2s \times 0,707 + 1 = \\ &= s^2 + 1,414s + 1 \end{aligned} \quad (1.62)$$

2 VÝBĚR KOMPONENT

Při výběru komponent jsou zohledněny určité požadavky. Mezi hlavní patří:

- mikropočítač s vestavěným A/Č převodníkem,
- paměť EEPROM pro uložení koeficientů filtru pro různé kmitočty,
- dostatečně velká flash paměť pro program,
- mikropočítač s hardwarovou násobičkou pro zrychlení výpočtů,
- klávesnice pro výběr alespoň ze čtyř možností.

2.1 MIKROPOČÍTAČ

Mikropočítač je integrovaný obvod, který se vyznačuje velkou spolehlivostí a kompaktností. Dělí se na dvě základní architektury, pro von Neumannovu architekturu je hlavním rysem společná paměť pro data i program. Pro přístup k paměti dat a programu není potřeba rozlišovat instrukce, tím se zjednoduší mikropočítač. Oba typy dat se přenáší po jedné datové sběrnici, z tohoto důvodu je přenos pomalejší. Při použití externích pamětí je to ale výhoda, dochází k redukci nutných počtů vstupů a výstupů. Harvardská architektura má oddělenou paměť dat a programu, kvůli nutnosti použití dvou sběrnic dochází k větší technologické náročnosti. Hlavní výhoda je možnost různé šířky pásma pro datové a programové sběrnice. Rozdělením sběrnic na datovou a programovou se zvětšuje rychlost vykonávání instrukcí, potřebná data i instrukce se dají číst v jeden okamžik. Častěji se využívá Harvardská architektura, u moderních architektur se může paměť jevit jako společná pro data i program, ale fyzicky jsou paměti připojeny větším počtem sběrnic (Bumba, 2011).

Hlavním faktorem pro výběr mikropočítače je výkon a kombinace periferních zařízení. Pro většinu vestavěných aplikací jsou dostačující osmibitové mikropočítače, které se používají již dlouho, a proto v nich je velký výběr. V případě potřeby většího výpočetního výkonu lze použít 16bitové nebo 32bitové mikropočítače. Mikropočítače mají různá periferní zařízení, většina z nich má několik vstupních a výstupních portů, časovačů a sériových rozhraní. Konfigurace pouzder a vývodů souvisí s počtem vstupů, výstupů a periferních zařízení. Při výběru mikropočítače se bere ohled i na velikost a umístění pamětí. Trendem se stávají mikropočítače, které mají tři typy pamětí. S pamětmi flash na uložení programu, EEPROM pro konfigurační data a SRAM na data. Dalším faktorem je kompatibilita s jinými řadami mikropočítačů, při programování ve vyšších jazycích, například jazyk C (Janeček, 1999).

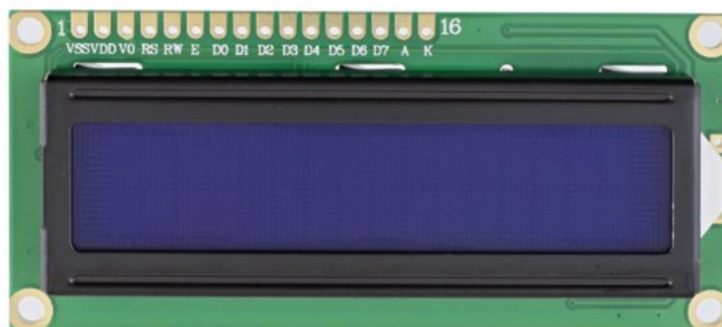
2.1.1 Mikropočítač ATmega644

ATmega644 patří mezi mikropočítače od Atmel AVR. AVR označuje skupinu 8bitových a 32bitových mikropočítačů s harvardskou architekturou a instrukční sadou typu RISC. Ta obsahuje hardwarovou násobičku a dalších 130 instrukcí. Použitím vhodného krystalu se může dosáhnout taktovacího kmitočtu až 20 MHz. Při této hodnotě může být výpočetní výkon až 20 MIPS, jedna instrukce se stihne vykonat za 50 ns. Registrové pole obsahuje 32 obecně použitelných registrů širokých 8 bitů (jeden bajt). Jejich označení je R0 až R31. Na přesun dat mezi registry nebo mezi paměti a registry je využita instrukční sada. Pro uložení konstant slouží stálá datová paměť EEPROM o velikosti 2 KB. Integrovaná paměť RAM s kapacitou 4 KB je vhodná pro realizaci bufferů nebo ukládání dat. Paměť flash s kapacitou 64 KB je použitelná pro uložení rozsáhlých programů. Ta se dá programovat pomocí sběrnice SPI, pro vývoj aplikací přímo v systému. Vestavěné periferie jsou ovládány pomocí vstupně výstupních registrů, mezi základní periferie patří:

- Napěťový komparátor – je realizován jako operační zesilovač bez zpětné vazby. Slouží k porovnání dvou hodnot napětí, které jsou přivedeny na invertující a neinvertující vstup, nebo ho lze využít pro vytvoření vstupní události čítače.
- Watchdog timer – je ochrana proti zabloudění programu ve smyčce. Watchdog je nezávislý čítač, musí být včas vynulován i za normálního chodu programu. Jinak resetuje procesor a uvede ho do výchozího stavu.
- Čítače/časovače – dva 8bitové a jeden 16bitový čítač/časovač, používají se pro odměření časových intervalů nebo pro čítání vnějších impulzů. 16bitový čítač/časovač může mít funkci PWM generátoru, dále můžou sloužit jako záchytný registr nebo výstupní komparátor.
- A/Č převodník – slouží k převodu analogového napětí na číslicové, aby ho mohl zpracovávat procesor. Je osmi kanálový 10bitový, doba převodu je 13 taktů, první převod je delší.
- Čtyři paralelní porty – vstupně/výstupní vývody pro přímé řízení vnějších periférií, každý vývod lze nastavit nezávisle. Pomocí funkce Pin Change se dají nastavit také jako vstupy vnějšího přerušení, reagující na změnu logické úrovně vstupu.
- Komunikační sběrnice – sběrnice USART, SPI a TWI (I²C) se dají připojit přídatné periferie, které používají sériové sběrnice pro komunikaci. Můžou sloužit i pro propojení mikropočítače se sériovým portem PC (Matoušek, 2013).

2.2 LCD DISPLEJ

LCD displej byl zvolen dvouřádkový s počtem 16 znaků na řádek. Obsahuje řadič HD44780, to je integrovaný obvod od firmy Hitachi, používaný pro ovládání maticových displejů. Každý znak se skládá z matice 5×8 bodů, osmý řádek se používá pro kurzor, takže vlastní znak zůstává 5×7 bodů. Řadič obsahuje znakovou sadu, kde je předdefinovaných 256 znaků včetně mezer, adresa znaků odpovídá jejich ASCII kódu. Dále umožňuje přidat osm vlastních znaků, jako jsou například písmena s diakritikou. Díky znakové sadě v řadiči se nemusí řídit rozsvícení jednotlivých pixelů, po napsání znaku řadič sám rozsvítí pixely v dané matici. Jako datová sběrnice slouží osm pinů LCD displeje. Data je možné přenášet využitím všech osmi datových vodičů, nebo pouze čtyř, kdy se zápis do LCD provádí nadvakrát. Pro řízení slouží řídicí signály RS, R/W a E. Displej je napájen stejnosměrným napětím 5 V. Podsvícení displeje má oddělené napájení od řadiče, aby se dalo v případě potřeby snížení spotřeby vypnout. Provozní teplota displeje je od $-20\text{ }^{\circ}\text{C}$ do $70\text{ }^{\circ}\text{C}$ (tajned.cz, 2014).



Obr. 2.1 – LCD alfanumerický displej

2.3 NAPÁJENÍ OBVODU

Napájení mikropočítačů neboli celého obvodu je řešeno pomocí spínaného napájecího zdroje typu SYS1421–0605. Poskytuje výstupní proud 1,2 A, stejnosměrné napětí 5 V a výkon 6 W. Napájecí konektor je typu $2,1 \times 5,5$ mm. Mikropočítač se dá připojit k PC pomocí sériového portu. V případě, že PC nemá sériový port, je nejjednodušším řešením pro připojení použít USB-RS 232 převodník.

Tab. 2.1 – Parametry zdroje napájení (www.tme.eu, nedatováno)

| Typ konektoru, mm | Výstupní napětí, V | Výstupní proud, A | Proudová ochrana, A | Přepětová ochrana, V |
|-------------------|--------------------|-------------------|---------------------|----------------------|
| 2,1 · 5,5 | 5 | 0 ~ 1,2 | 1,5 ~ 3,5 | 12 ± 4 |
| mini USB | 5 | 0 ~ 1,2 | 1,3 ~ 3,0 | 12 ± 4 |
| micro USB | 5 | 0 ~ 1,2 | 1,3 ~ 3,0 | 12 ± 4 |

2.4 A/Č PŘEVODNÍK

Mikropočítač ATmega644 obsahuje vlastní A/Č převodník, díky kterému je mikropočítač univerzálnější. Není nutné připojovat externí obvody pro digitalizaci analogových veličin.

A/Č převodník je 10bitový a je založen na principu postupné aproximace. Pomocí analogového multiplexeru, na který je A/Č převodník připojen, je možné snímat napětí až osmi vstupů, zabudované reference a analogové nuly. Při převodu musí být hodnota vstupního napětí stabilní, z toho důvodu má A/Č převodník vzorkovací obvod spojený se zesilovačem. Napájecí napětí AU_{CC} je odděleno od analogové země $AGND$. $AGND$ musí být spojený s vývodem země GND pouze v jediném bodě, na vývodu napájecího napětí musí být napětí $U_{CC} \pm 0,3$ V (Matoušek, 2013).

Základní parametry A/Č převodníku:

- rozlišení až 10 bitů,
- rychlost převodu až 15000 vzorků za sekundu při 10bitovém rozlišení,
- interní referenční napětí 1,1 V a 2,56 V,
- tři různé režimy převodu,
- přerušení po dokončení převodu,
- absolutní chyba ± 2 LSB,
- osm multiplexovaných vstupů v provedení SE,
- dva diferenční vstupy,
- v režimu Idle potlačovač šumu (Matoušek, 2013).

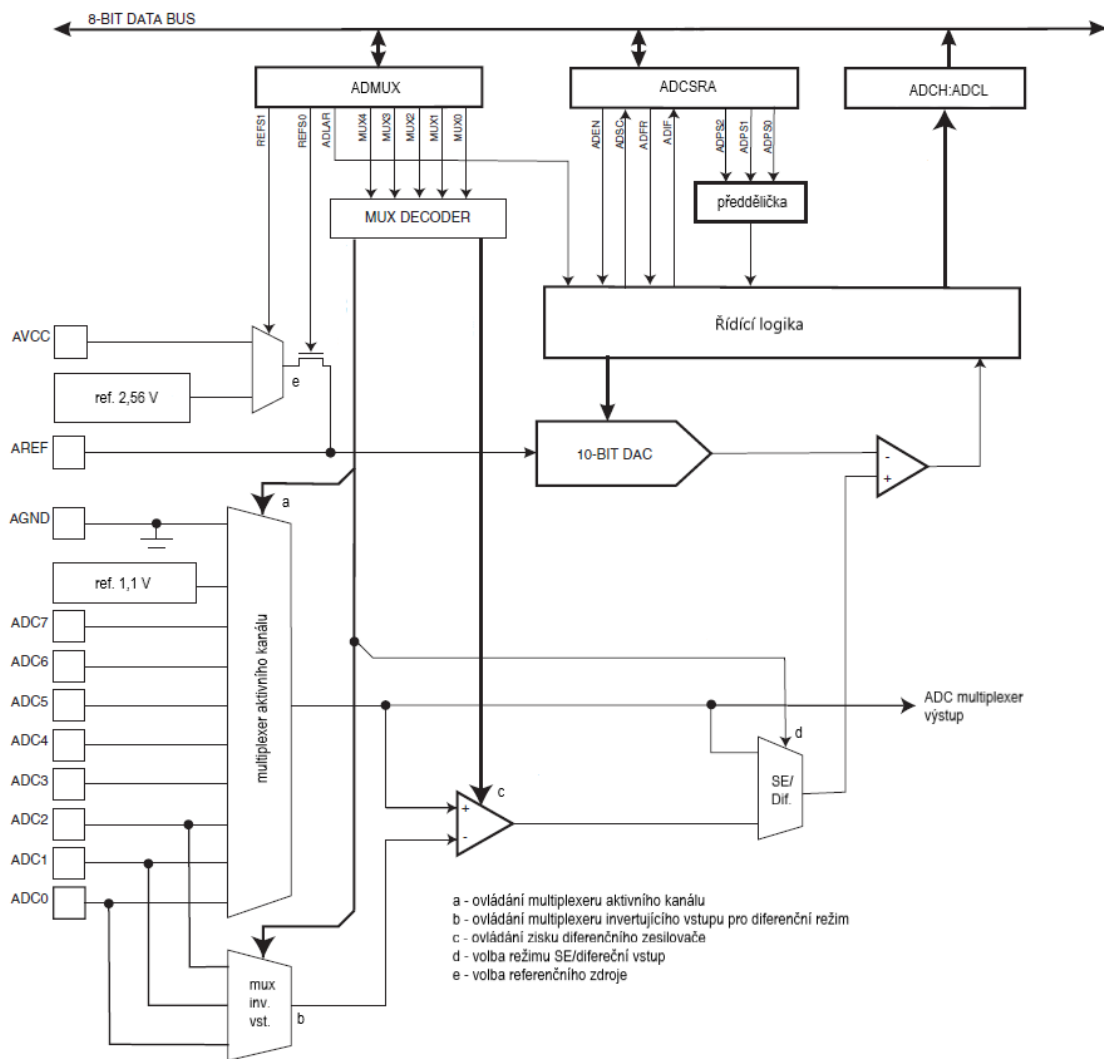
Tab. 2.2 – Parametry A/Č převodníku (Matoušek, 2013)

| Označení | Význam | Minimum | Typ | Maximum | Jednotky |
|------------|------------------------|------------------|-----|------------------|------------|
| t_{CONV} | Doba převodu | 65 | | 260 | μ s |
| f_{ADC} | Hodinový kmitočet | 50 | | 1000 | kHz |
| AU_{CC} | Napětí analogové části | $U_{CC} - 0,3$ V | | $U_{CC} + 0,3$ V | V |
| $AREF$ | Referenční napětí | 2 | | AU_{CC} | V |
| U_{IN} | Vstupní napětí | $AGND$ | | $AREF$ | V |
| R_{IN} | Vstupní odpor | | 100 | | M Ω |

2.4.1 Popis činnosti A/Č převodníku

Pomocí postupné aproximace dochází k převodu vstupního analogového napětí na 10bitovou číslicovou hodnotu. Maximum lze vybrat mezi napětím přivedeným na vývod *AREF*, vnitřním referenčním napětím 1,1 V a 2,56 V a napětím vývodu *AUCC*. Výběr se provádí v registru *ADMUX*. Minimum odpovídá hodnotě na vývodu *AGND*. Vnější zdroj referenčního napětí je citlivější na rušení než vnitřní zdroj referenčního napětí. Při výběru libovolného napětí musí být na vývodu *AREF* připojen blokovací kondenzátor.

Zápisem bitů *MUX0* až *MUX4* do registru *ADMUX* se zvolí vstupní analogové kanály. Kanály se dají nastavit jako *SE*, tzn. napětí se měří proti *AGND*. Do diferenčního zesilovače se dají přivést některé vývody A/Č převodníku jako invertující nebo neinvertující vstupy. Pokud jsou vybrány diferenční kanály, do A/Č převodníku vstupuje rozdíl napětí daných vývodů nebo zesílení. U kanálů typu *SE* se zesílení nastavit nedá. Z důvodu snížení spotřeby mikropočítače je v defaultním stavu A/Č převodník vypnut. Zapíná se nastavením bitu *ADEN* v registru *ADCSRA* na hodnotu *ADEN* = 1. Výsledek 10bitového převodu se ukládá do registrového páru *ADCH:ADCL*. Čtení se začíná z registru *ADCL*, když je přečten registr *ADCL*, je povolen přístup k registru *ADCH*. Registr *ADCL* se nedá číst dvakrát za sebou, došlo by ke ztrátě výsledku převodu bez přečtení registru *ADCH* (Matoušek, 2013).

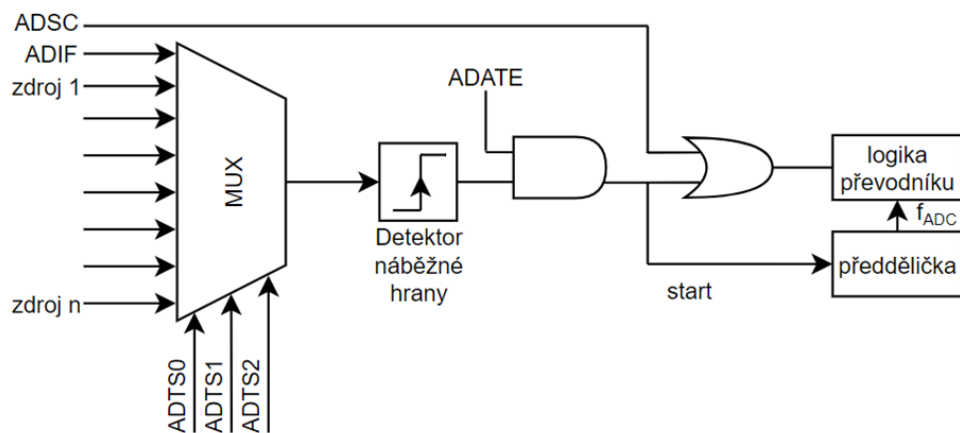


Obr. 2.2 – Blokové schéma A/D převodníku (Matoušek, 2013)

2.4.2 Režimy startu převodu

Začátek převodu může být nastaven třemi způsoby:

- Volný běh – převod probíhá automaticky, zdrojem je příznak ADIF. Ten signalizuje konec předchozího převodu. Pro začátek převodu se musí nastavit bit ADSC, potom už probíhá převod automaticky, nezávisle na vynulování příznaku ADIF.
- Jednoduchý převod – převod se začne nastavením bitu ADSC, který se musí nastavit při každém novém převodu. Bit zůstane nastaven až do konce převodu, potom dochází k hardwarovému vynulování. Při změně kanálu během převodu, dojde ke změně až po dokončení současného převodu.
- Automatické spouštění – nastaví se bitem ADATE, bity ADTS0 až ADTS2 se vybere spouštěcí zdroj. Náběžnou hranou spouštěcího signálu dochází ke startu převodu. Stávající převod se nepřerušuje novou náběžnou hranou, je ignorována. Pro automatický běh musí být nastaven příznak přerušování ADIF. Před začátkem nového převodu musí být vynulován, aby vznikla náběžná hrana (Matoušek, 2013).

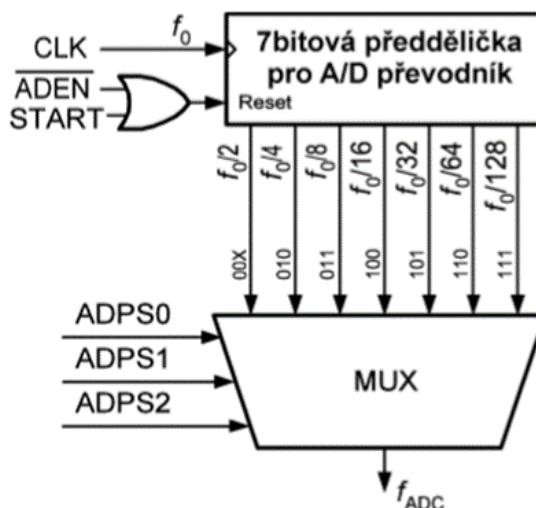


Obr. 2.3 – Spouštěcí logika A/D převodníku (Matoušek, 2013)

2.4.3 Předdělička A/Č převodníku

Pomocí předděličky se mění systémové hodiny na hodinový kmitočet f_{ADC} , který je přijatelný pro A/Č převodník, viz obr. 2.4. To musí být provedeno pro maximální přesnost postupné aproximace. K získání maximální přesnosti musí být vstupní hodinový kmitočet v rozsahu 50 až 200 kHz. Výběr hodinového kmitočtu se provádí spodními třemi bity ADPS0 až ADPS2 (Brtník, Matoušek, 2010).

Čítání předděličky se začne nastavením bitu ADEN, je aktivní, dokud není bit ADEN vynulován. Běžný převod trvá 13 hodinových cyklů f_{ADC} , první převod trvá 25 hodinových cyklů f_{ADC} . Během prvního převodu dochází k inicializaci. Vzorkování při prvním převodu zabere 13,5 hodinových cyklů f_{ADC} a 1,5 hodinových cyklů f_{ADC} při běžném převodu (Matoušek, 2013).



Obr. 2.4 – Předdělička pro A/D převodník (Brtník, Matoušek, 2010)

2.4.4 Uložení výsledku převodu

Nastavením bitu ADIF dojde k ukončení převodu. Výsledek je uložen do registrového páru ADCH:ADCL, odkud ho lze i číst.

Pro SE vstupy je hodnota 0x0000 analogová zem, hodnota 0x03FF tj. 1023 představuje vybrané referenční napětí bez nejméně významného bitu. Platí vztah:

$$ADCH:ADCL = 1024 \times \frac{U_{VST}}{U_{REF}} \quad (2.1)$$

kde U_{VST} – vstupní napětí, V,

U_{REF} – referenční napětí, V.

U diferenčních vstupů se výsledek zobrazuje jako dvojkový doplněk od 0x0200 tj. -512 až po 0x01FF tj. 511. Platí vztah:

$$ADCH:ADCL = G \times 512 \times \frac{(U_{POS} - U_{NEG})}{U_{REF}} \quad (2.2)$$

kde U_{POS} – napětí neinvertujícího vstupu, V,

U_{NEG} – napětí invertujícího vstupu, V,

U_{REF} – referenční napětí, V,

G – zisk, dB (Matoušek, 2013).

2.4.5 Změna kanálu

Přepnutí kanálu je možné po dokončení převodu, pro ukončení převodu se používá rutina přerušení. Nový převod začíná hned po aktivaci přerušení. Změna analogového kanálu v multiplexeru v obslužné rutině je aktivní až na třetí převod. To znamená, že změna analogového vstupu je zpožděna o dokončení stávajícího převodu. Při režimu volného běhu se nemusí použít rutina přerušení pro ukončení převodu. Režim volného běhu dokáže sledovat více kanálů najednou. Diferenční kanál se může změnit až po ustálení výstupu diferenčního zesilovače. To trvá 125 μ s, potom může začít nový převod (Matoušek, 2013).

2.4.6 Registry

Řídicí **registr ADMUX** slouží pro výběr analogového kanálu:

- **REFS1, REFS0** – výběr referenčního zdroje napětí pro A/Č převodník.
- **ADLAR** – zarovnání výsledku do registrů ADCH:ADCL. Pokud **ADLAR** = 0 jsou bity po převodu umístěny v registru ADCL a nejvýše dva bitvy ve spodních bitech registru ADCH. Při zarovnání výsledku doleva, **ADLAR** = 1 a osm nejvyšších bitů po převodu je uloženo do registru ADCH, zbylé dva bity se uloží do nejvyšších dvou bitů registru ADCL.
- **MUX0** až **MUX4** – nastavení zesílení, volba vstupního kanálu, výběr mezi SE a diferenčním vstupem viz obr. 2.5 (Matoušek, 2013).

Stavový řídicí **registr ADCSRA**:

- **ADEN** – zapnutí a vypnutí A/Č převodníku, vypnutím A/Č převodníku při probíhajícím převodu se převod předčasně ukončí.
- **ADSC** – start převodu se začne nastavením bitu **ADSC** = 1, v režimu jednoduchého převodu se musí bit nastavit pro spuštění každého převodu. Pro režim volného běhu se nastavuje pouze při prvním spuštění. Po dokončení převodu se bit hardwarově vynuluje, při probíhajícím převodu nemá programové vynulování bitu žádný vliv na převod.
- **ADATE** – možnost automatického spuštění převodu nastavením hodnoty **ADATE** = 1. Při automatickém spuštění musí nejdříve dojít k aktivaci zdroje přerušení.
- **ADIF** – je příznak přerušení A/Č převodníku. Příznak se aktivuje po dokončení převodu a aktualizaci registrů ADCH:ADCL kam se výsledek uloží.

- **ADIE** – povolení přerušení A/Č převodníku po dokončení převodu.
- **ADPS0 až ADPS2** – výběr hodinového kmitočtu pomocí předděličky A/Č převodníku, viz obr. 2.6 (Váňa, 2003).

Registr DIDR0 se používá k odstavení digitálních vstupů. Nastavením jednoho či více bitů ADC7D až ADC0D se odstaví vstupní digitální buffer. Potom digitální buffer přestane reagovat na měnící se analogový signál a sníží se jeho spotřeba.

V **registru ADCSRB** se vybírá zdroj automatického spouštění. K aktivaci dojde vybráním jednoho z bitů ADTS2 až ADTS0 a nastavením bitu ADATE. Pro spuštění musí přijít náběžná hrana zvolené podmínky.

| | | | | | | | | | |
|---------------|-------|-------|-------|------|------|------|------|------|-------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | REFS1 | REFS0 | ADLAR | MUX4 | MUX3 | MUX2 | MUX1 | MUX0 | ADMUX |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Obr. 2.5 – K registru ADMUX (Váňa, 2003)

| | | | | | | | | | |
|---------------|------|------|-------|------|------|-------|-------|-------|--------|
| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| | ADEN | ADSC | ADATE | ADIF | ADIE | ADPS2 | ADPS1 | ADPS0 | ADCSRA |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| Initial Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

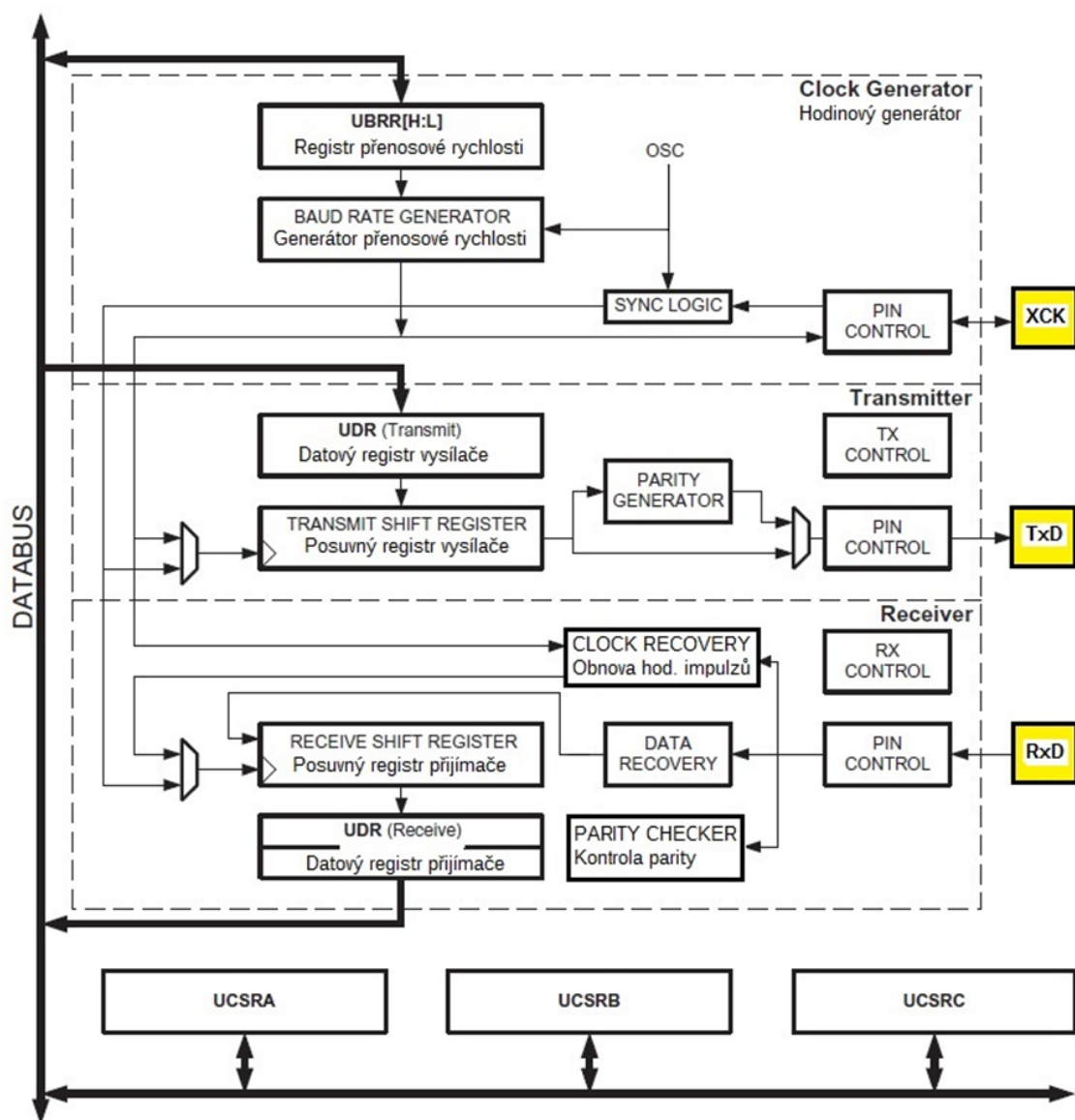
Obr. 2.6 – K registru ADCSRA (Váňa, 2003)

2.5 JEDNOTKA USART

USART realizuje sériovou komunikaci mikropočítače s periferií nebo počítačem. Napěťové úrovně jsou jiné než u rozhraní RS-232. Mikropočítače AVR používají pro jednotku USART stejné napěťové úrovně jako pro logiku TTL. Z toho důvodu se mezi jednotku USART a rozhraní RS-232 zapojuje převodník napěťových úrovní. Na výběr je ze čtyř komunikačních režimů, normálního asynchronního režimu a asynchronního režimu s dvojnásobnou rychlostí. Synchronní režimy se dělí na režimy master a slave (Hrbáček, 2002).

Na obr. 2.7 je blokové schéma jednotky USART, dělí se na části:

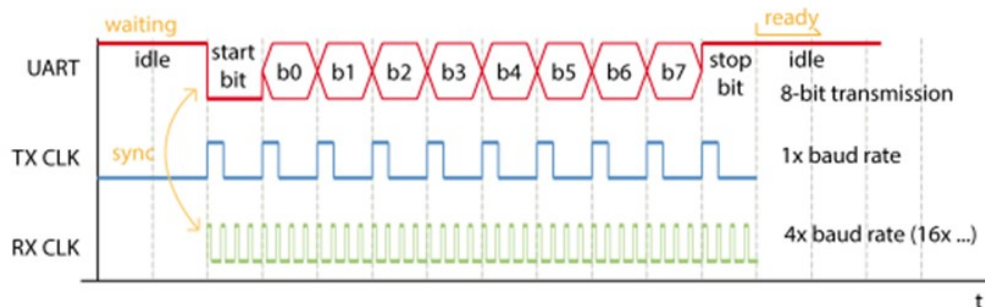
- Hodinový generátor – pro synchronní sériový režim je potřeba synchronizační logika pro vnější hodinový vstup.
- Vysílač (transmitter) – obsahuje jeden zapisovací buffer, sloužící pro přenos dat bez zpoždění mezi rámci bez zpoždění. Jeho součástí je také sériový posuvný registr, generátor parity a řídicí jednotka pro obsluhu sériových rámců.
- Přijímač (receiver) – nejsložitější část jednotky, obsahuje jednotky pro obnovu dat a hodin. Tyto jednotky slouží pro příjem asynchronních dat. Přijímač podporuje stejné formáty rámců jako vysílač, obstarává i detekci chyb. Dále obsahuje posuvný registr, kontrolu parity a datový registr přijímače (Matoušek, 2013).



Obr. 2.7 – Blokové schéma jednotky USART (Hrbáček, 2002)

Jednotka USART pracuje v synchronním i asynchronním režimu. XCK je vývod pro použití synchronního režimu, slouží přenosu hodinových synchronizačních impulzů na sběrnici. Synchronní a asynchronní režim se přepíná pomocí dvou samostatných multiplexerů. Registry UCSRA, UCSRB, UCSRC jsou řídicí pro jednotku USART. Rychlost přenosu sériového rozhraní nastavuje registrový pár UBRRL:UBRRH. Posuvné registry ve vysílači a přijímači jsou základem jednotky. Ve vysílači jsou pomocí posuvného registru převáděna paralelní data z datového registru UDR na sériový přenos. Přijímač současně převádí sériový datový přenos na paralelní data posuvným registrem, ty jsou zapisována do registru UDR (Hrbáček, 2002).

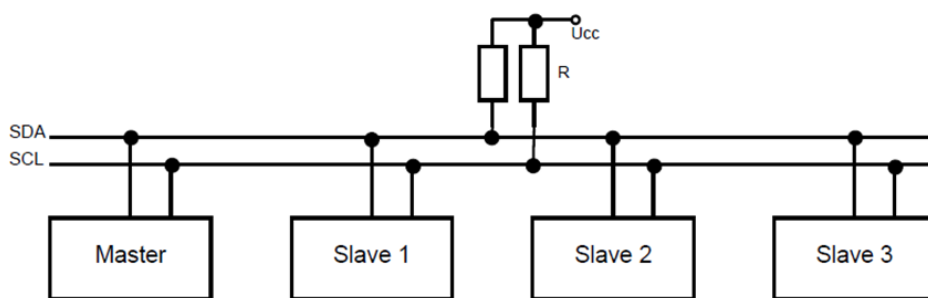
Obecné schéma přenosu je zobrazeno na obr. 2.8. Data se přenášejí po jednotlivých znacích, jednotlivé bity se přenášejí sériově. Z důvodu rozeznání odpojené sběrnice a klidového stavu, je sběrnice v klidu ve stavu log. 1. Přenos začíná start-bitem, který má stav log. 0, následují datové bity. Nejčastěji je datových bitů osm, jsou v pořadí od nejnižšího po nejvyšší bit. Vysílání se ukončí stop-bitem, který má stejný stav jako klidový stav. Vysílaná data mohou být volitelně doplněna o paritní bit (src.athaj.cz, nedatováno).



Obr. 2.8 – Časování přenosu pomocí USART (src.athaj.cz, nedatováno)

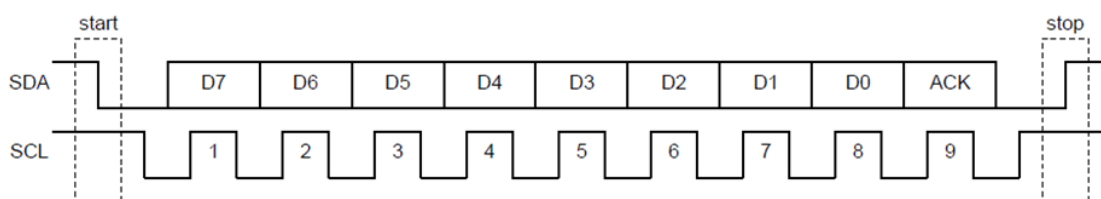
2.6 JEDNOTKA TWI

TWI je dalším typem sériového rozhraní, je známé taky pod názvem I²C. Pomocí této sběrnice lze připojit až 128 zařízení, každé je připojeno dvěma vodiči. Jednotka TWI funguje jako half-duplex, tzn. může vysílat pouze jedno zařízení v daný okamžik, ale sběrnice je obousměrná (zavavov.cz, 2014).



Obr. 2.9 – Ukázka zapojení sběrnice TWI (zavavov.cz, 2014)

Na sběrnici je jeden řídicí obvod master a více podřízených obvodů slave. Přenos dat zahajuje master posláním sedmibitové adresy pro výběr slave zařízení a log. 1 v osmém bitu. Po přijetí potvrzovacího bitu ACK vyšle master data po vodiči SDA. Pro příjem dat zahajuje komunikaci master opět zasláním adresy pro výběr obvodu slave, ale v osmém bitu bude log. 0. Průběh dat je vidět na obr. 2.10, synchronizace přenosu probíhá na vodiči SCL impulsy s frekvencí 10 kHz až 400 kHz (zavavov.cz, 2014).

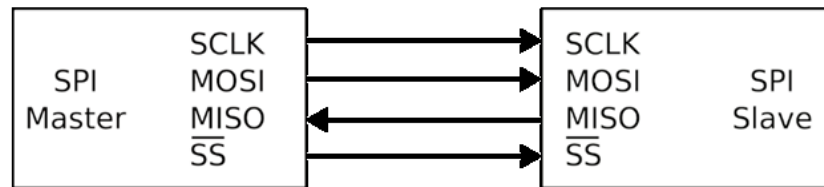


Obr. 2.10 – Průběh přenosu dat na linkách SDA a SCL (zavavov.cz, 2014)

2.7 SÉRIOVÝ KANÁL SPI

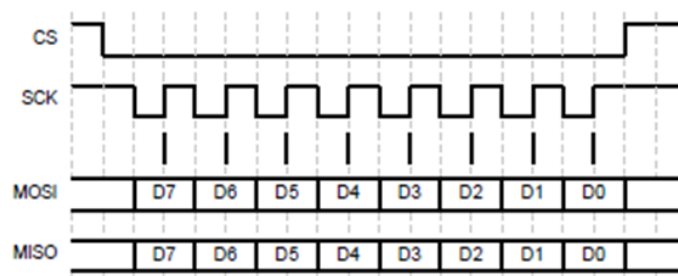
Sériový kanál realizuje vysokorychlostní synchronní komunikaci mezi mikropočítači a periferiemi. Komunikace je obousměrná, schopnost současně přijímat i vysílat (full-duplex). Jak je vidět z obr. 2.11 zařízení může fungovat jako master, tj. řídicí obvod, nebo slave tj. řízený

obvod. Vybrat lze ze sedmi programovatelných rychlostí, maximální rychlost přenosu je 10 Mb/s (root.cz, 2008).



Obr. 2.11 – Princip komunikace po sběrnici SPI (root.cz, 2008)

Zápis dat do registru začíná hodinový generátor, data z obvodu master jsou vysílána vývodem MOSI. Obvod slave data přijímá také vývodem MOSI. Synchronizace přenosu bitů probíhá pomocí hodinového signálu SCLK. Po odeslání celého bajtu se generátor hodin zastaví a nastaví se příznak konce. Pomocí vývodu \bar{SS} se vybírá zařízení slave, při výběru je vývod nastaven na nulu. Pokud je nastaven na jedničku, SPI port je neaktivní a vývod MOSI se dá použít jako obyčejný vývod (Matoušek, 2013).



Obr. 2.12 – Průběh přenosu dat po sběrnici SPI (root.cz, 2008)

3 REALIZACE HARDWARU

Pro realizaci obvodu jsem si vybíral ze dvou navržených schémat. Na obr. 3.2 je schéma s dvěma fyzickými displeji, které by bylo dražší pro realizaci kvůli většímu počtu součástek. Pro realizaci bylo použito schéma na obr. 3.1, které bylo jednodušší při realizaci DPS, ale programátorsky delší. Displej je na DPS pouze jeden, střídavě ukazuje kmitočet a napětí po průchodu filtrem.

Antialiasingový filtr je realizován prostým RC členem, chovajícím se jako dolní propust, zapojeným mezi vstupní konektor a vstup do obvodu. Jeho mezní kmitočet je: $f_{mez} = 5$ kHz, kapacitor volím 100 nF, takže odpor se spočítá pomocí vztahu (3.1).

$$R = \frac{1}{2\pi f_{mez} C} = \frac{1}{2\pi \times 5000 \times 100 \times 10^{-9}} = 318,47 \quad (3.1)$$

kde R – odpor RC členu, Ω ,

f_{mez} – mezní kmitočet, Hz,

C – kapacitor RC členu, F.

Odpor byl vybrán zaokrouhlením na hodnotu z řady E192, aby byl mezní kmitočet co nejpřesnější.

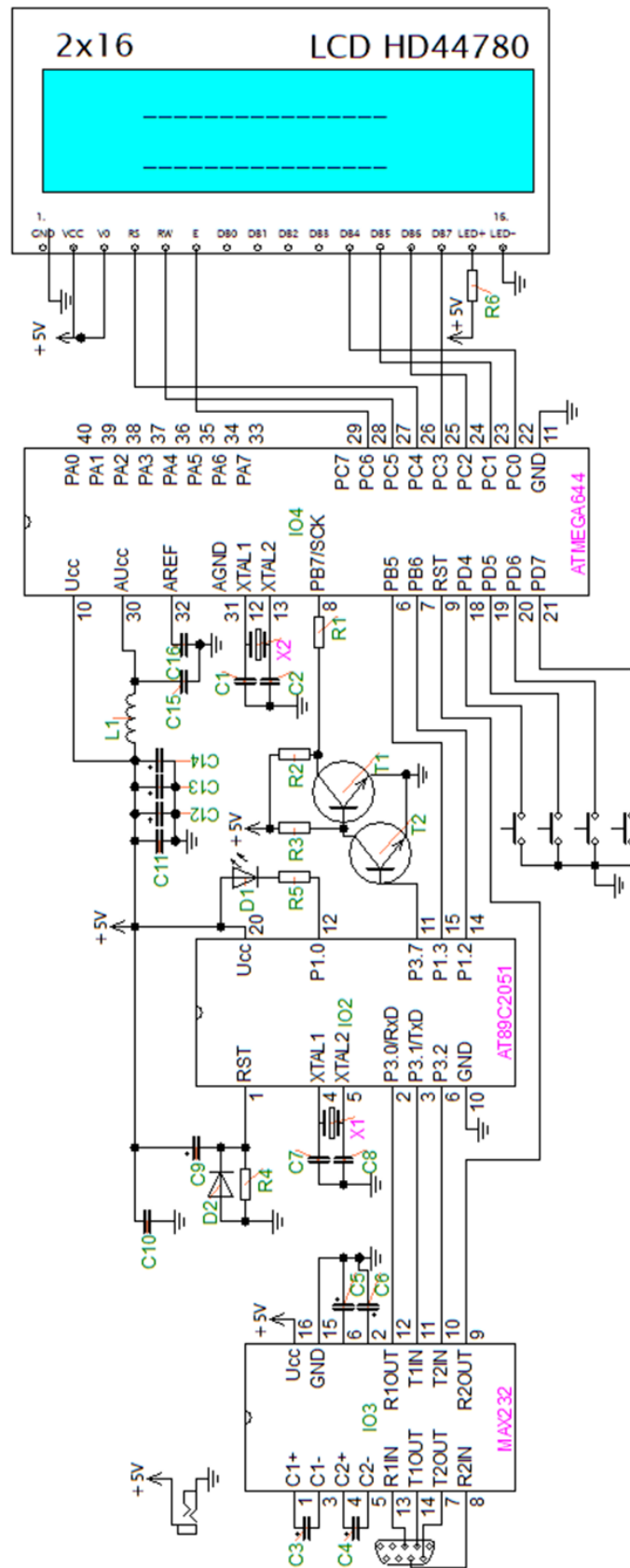
Obvod je napájen pomocí vnějšího spínaného adaptéru 5 V. Pro propojení adaptéru s obvodem slouží konektor CON1.

K propojení s počítačem slouží sériový port (RS-232). V současné době se již moc nepoužívá, je nahrazován výkonnějším univerzálním sériovým rozhraním (USB). Pro připojení k počítači s absencí RS-232 lze využít převodník USB/RS-232.

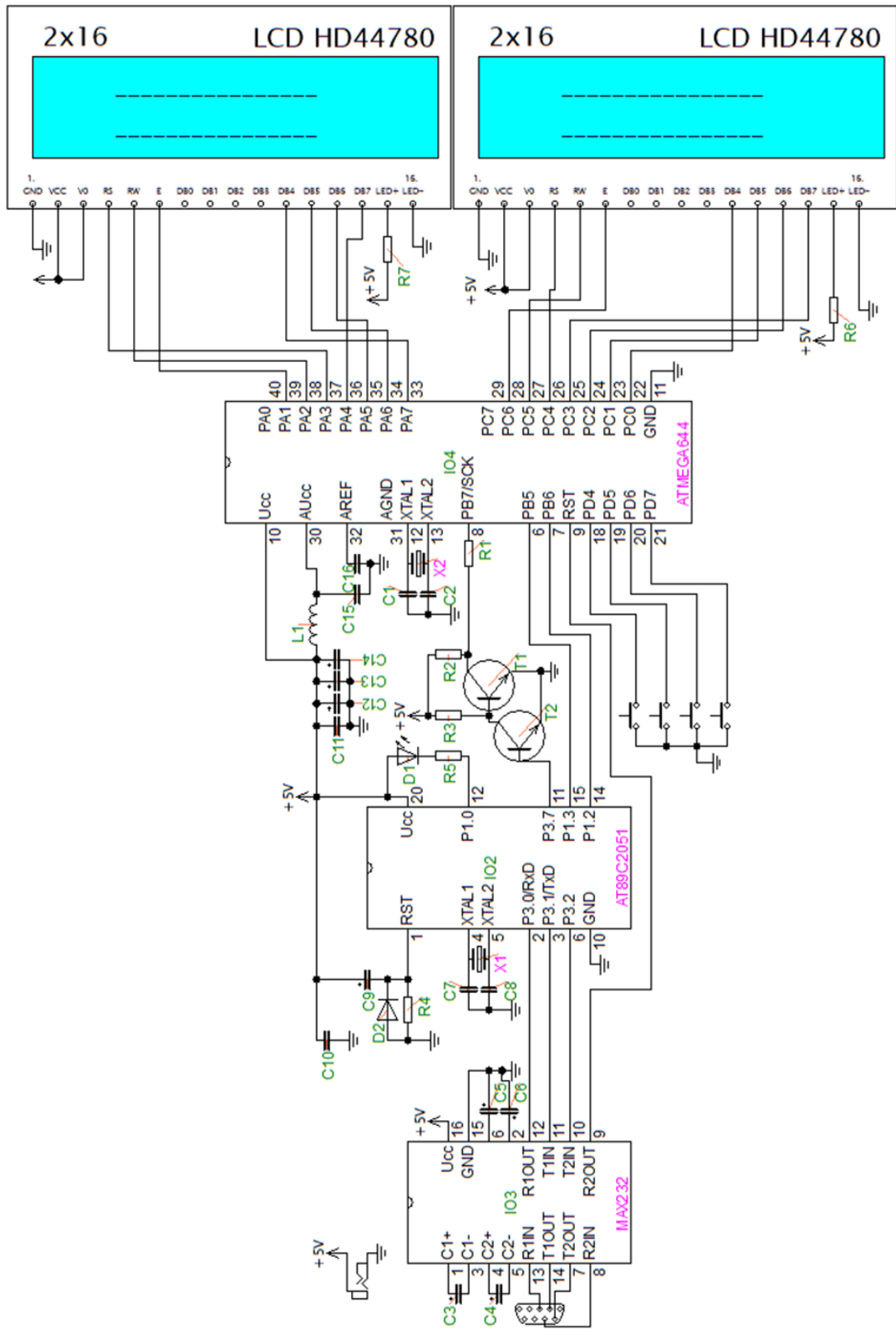
MAX232 je integrovaný obvod sloužící pro převod napětíových úrovní. RS-232 se převádí na TTL úroveň. Pro napájení tohoto IO stačí pouze jeden zdroj napětí a to +5 V. Obsahuje pouze dva převodníky, nelze převést všechny řídicí signály. V tomto případě je možné převádět pouze signály Rx, Tx, RTS a CTS. Pro převod všech řídicích signálů musí být použita dvojice obvodů MAX232. Čtveřice kondenzátorů slouží k správné funkci násobiče napětí z +5 V na +10 V a invertoru napětí z +10 V na -10 V.

Zapojení LCD displeje je řešeno zapojením pouze čtyř datových pinů DB4 – DB7. Data se rozdělí na dvě čtveřice bitů, které jsou poslány za sebou. Zbylé čtyři datové piny zůstanou nezapojené. Řídicí pin RS rozlišuje, jestli se jedná o data nebo instrukce. Pokud jsou přenášeny instrukce, je pin RS v log. 0, když je RS v log.1, jsou přenášeny znaky k zobrazení, případně

k uložení do paměti. Zápis dat do LCD a čtení dat z LCD je řízeno pinem R/W. Pomocí pinu E je vzorkovaná komunikace sestupnou hranou signálu.



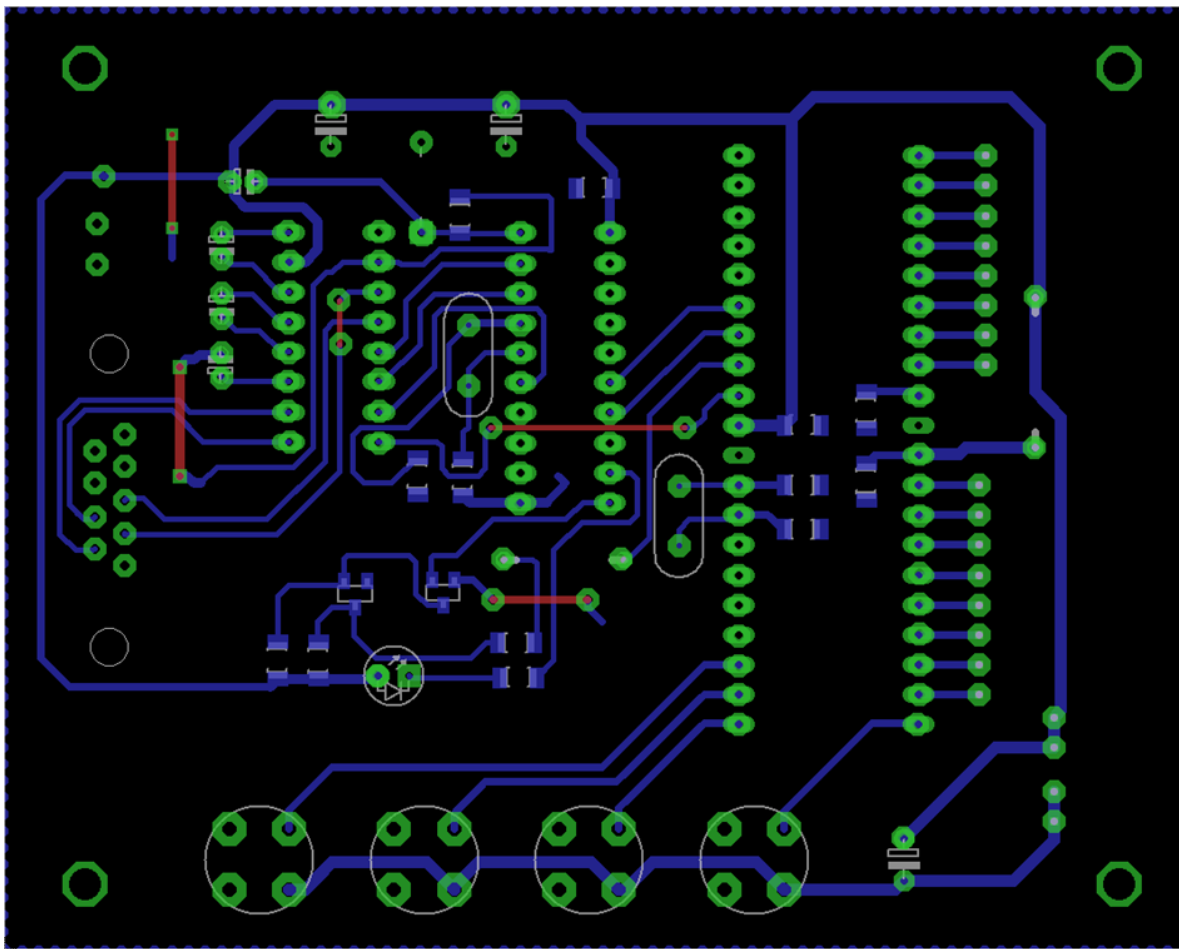
Obr. 3.1 – Schéma realizovaného obvodu



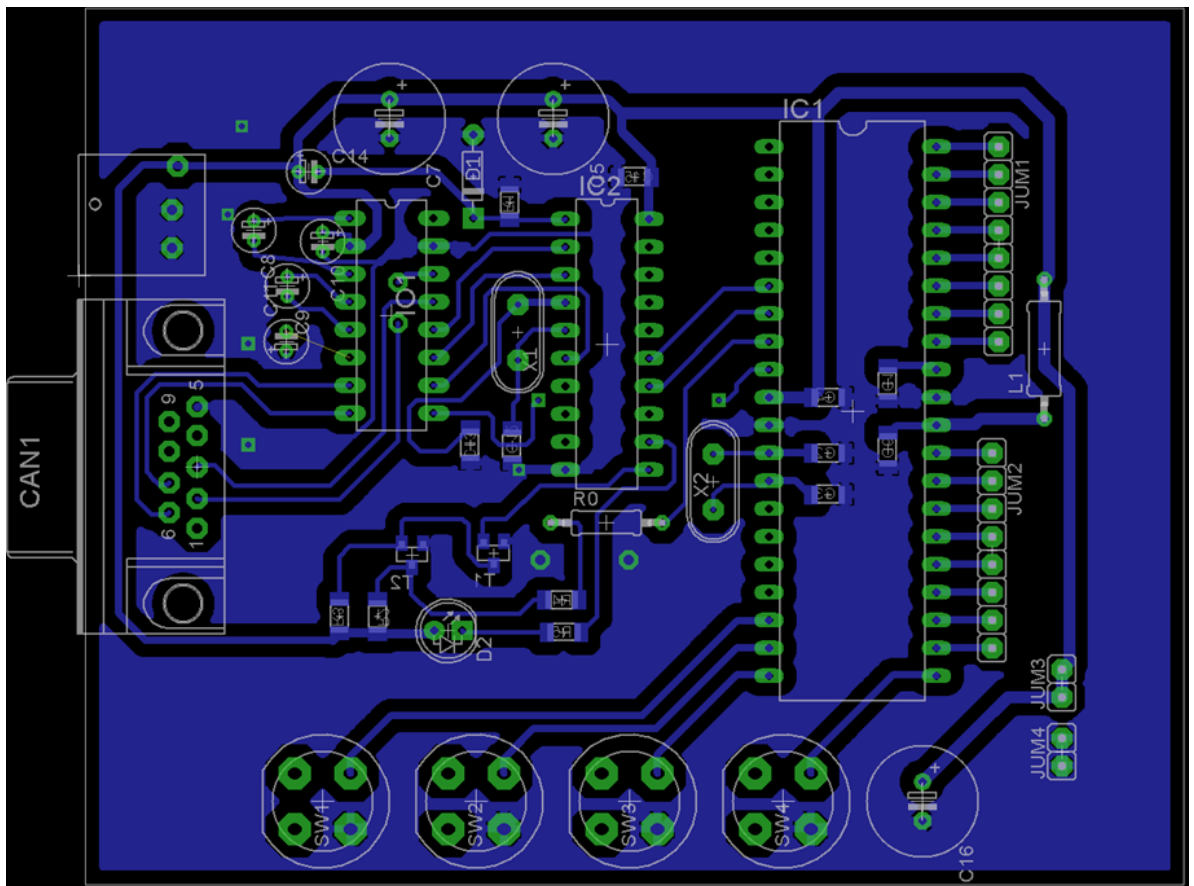
Obr. 3.2 – Schéma se dvěma fyzickými displeji

3.1 NÁVRH DPS

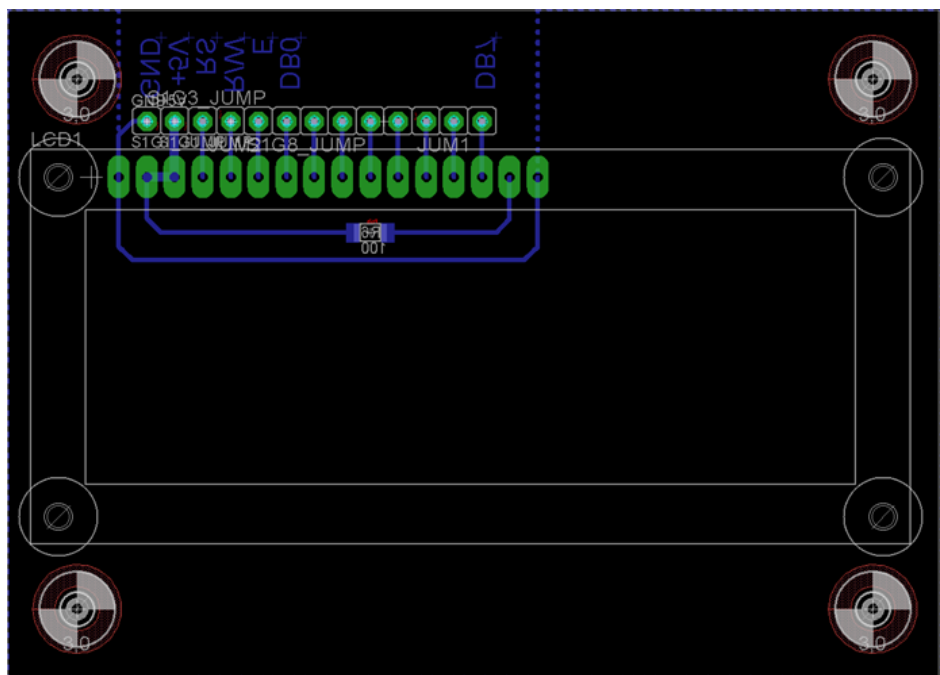
Návrh desky plošných spojů je vytvořen pomocí softwaru Eagle jako jednostranný. Z důvodu omezení ve verzi Eagle Light na maximální velikost desky s jednou stranou 80 mm a druhou 100 mm musela být deska plošných spojů rozdělena na dvě menší. Na dalších stranách jsou zobrazeny osazovací plány DPS i se stranou spojů a vylitým polygonem, následují obrázky osazené DPS.



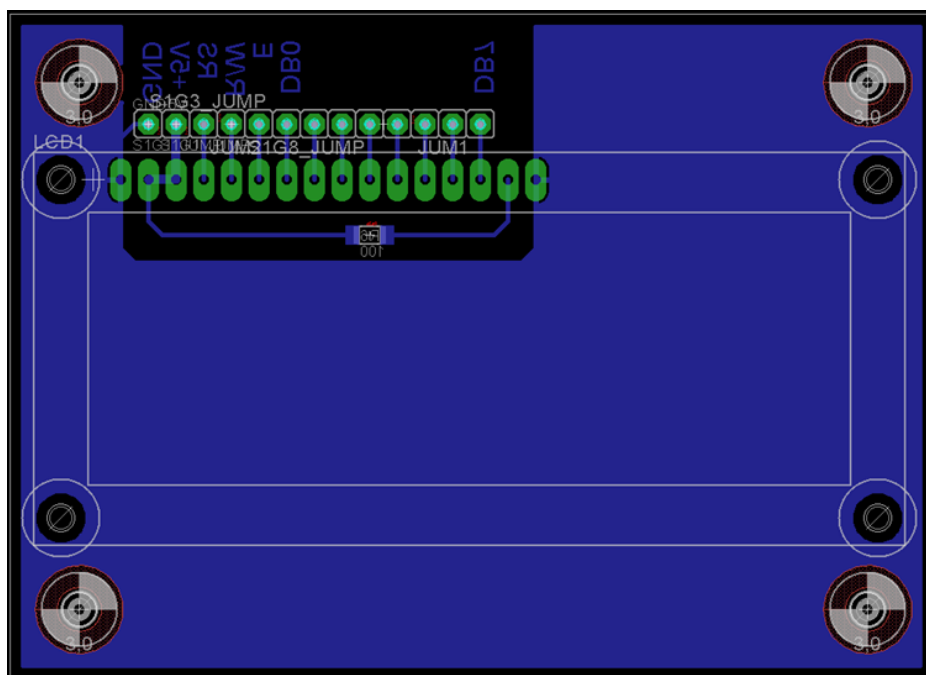
Obr. 3.3 – DPS selektivního voltmetru



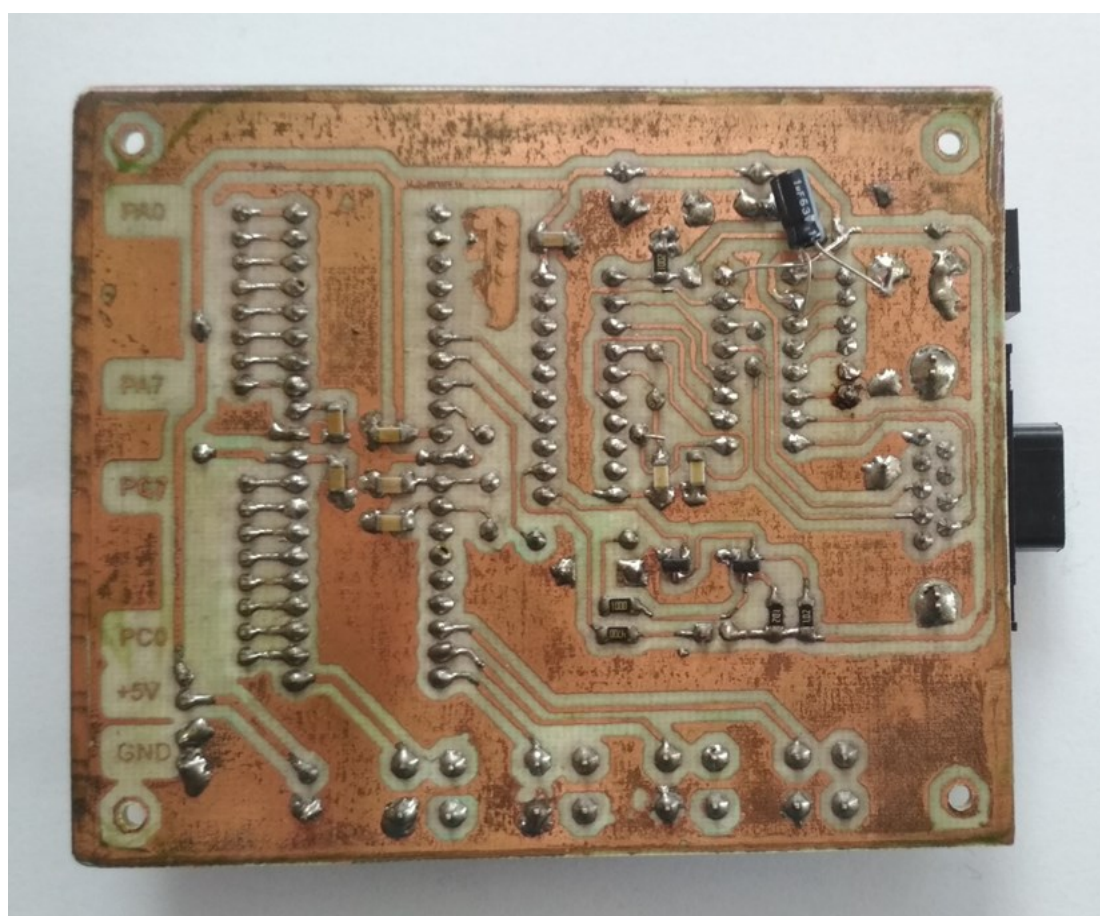
Obr. 3.4 – DPS selektivního voltmetru s vylitým polygonem



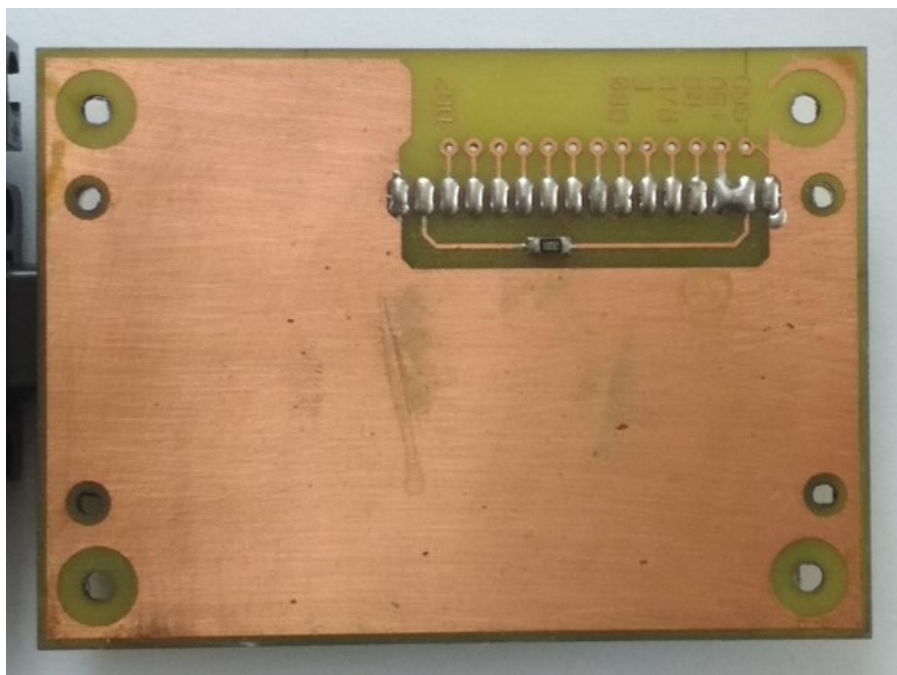
Obr. 3.5 – DPS LCD displeje



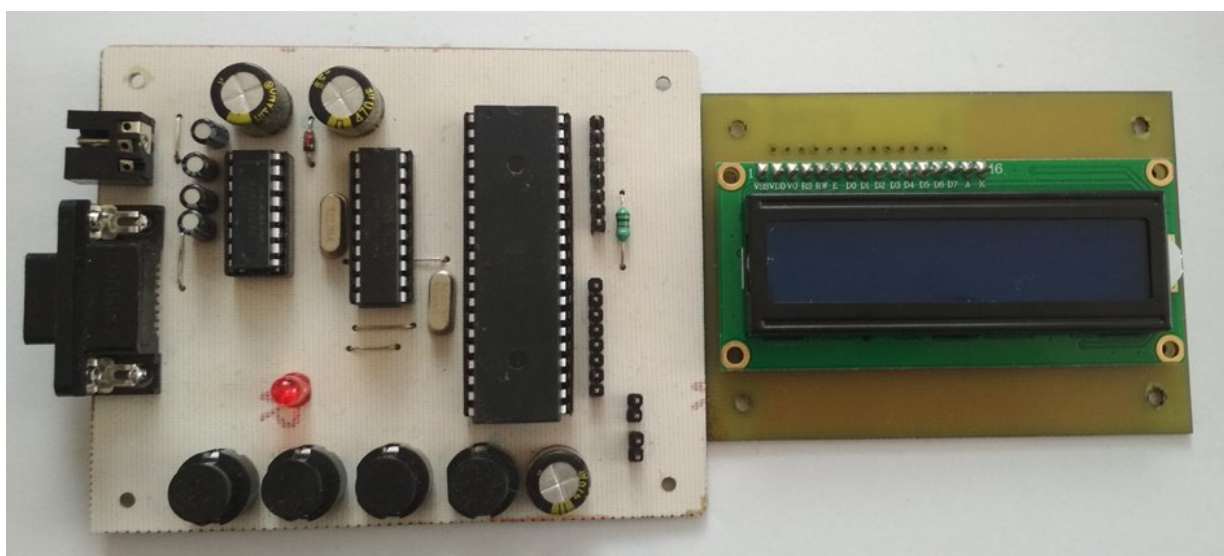
Obr. 3.6 – DPS LCD displeje s vylitým polygonem



Obr. 3.7 – Osazená deska selektivního voltmetru (strana SMD součástek)



Obr. 3.8 – Osazená deska LCD displeje (strana SMD součástek)



Obr. 3.9 – Osazená deska selektivního voltmetru s LCD displejem

4 REALIZACE SOFTWARE

4.1 NÁVRH ČÍSLICOVÉHO FILTRU

Úkolem je navrhnout přeladitelný číslicový filtr typu pásmová propust pro pásmo kmitočtů 500 Hz až 3000 Hz. K výpočtu jsem využil bilineární transformaci, která má maximálně plochou kmitočtovou charakteristiku, tedy s Butterworthovou aproximací. Vzorkovací frekvence je 10000 Hz.

Řešení:

Přepočet kmitočtů:

$$T_{vz} = \frac{1}{f_{vz}} = \frac{1}{10000} \quad (4.1)$$

$$\Omega_{c1} = \frac{2}{T_{vz}} \times \operatorname{tg} \frac{2\pi f_{c1} T_{vz}}{2} = 20000 \times \operatorname{tg} \frac{1000\pi}{20000} = 3168 \quad (4.2)$$

$$\Omega_{c2} = \frac{2}{T_{vz}} \times \operatorname{tg} \frac{2\pi f_{c2} T_{vz}}{2} = 20000 \times \operatorname{tg} \frac{6000\pi}{20000} = 27528 \quad (4.3)$$

kde T_{vz} – vzorkovací perioda, s,

f_{vz} – vzorkovací frekvence, Hz,

f_{c1} – mezní kmitočet, Hz,

f_{c2} – mezní kmitočet, Hz,

Ω_{c1} – mezní úhlový kmitočet, $\text{rad} \cdot \text{s}^{-1}$,

Ω_{c2} – mezní úhlový kmitočet, $\text{rad} \cdot \text{s}^{-1}$.

Výpočet šířky pásma:

$$\Delta\Omega = \Omega_{c2} - \Omega_{c1} = 27528 - 3168 = 24360 \quad (4.4)$$

kde $\Delta\Omega$ – šířka pásma

Ω_{c1} – mezní úhlový kmitočet, $\text{rad} \cdot \text{s}^{-1}$,

Ω_{c2} – mezní úhlový kmitočet, $\text{rad} \cdot \text{s}^{-1}$.

Uvažoval jsem filtr 2. řádu, podle řádu filtru se pozná počet článků filtru. Jednotlivé koeficienty se odečtou z tab. 4.1 pro daný řád filtru.

Tab. 4.1 – Koeficienty pro filtr druhého řádu

| n | a0 | a1 | a2 |
|---|----|--------|----|
| 2 | 1 | 1,4142 | 1 |

Filtrační funkce v oblasti $s = j\omega$ dosazením do obecného vztahu (4.4) bude

$$F(s) = \frac{1}{\sum_{i=0}^N a_i s^i} \quad (4.5)$$

$$F(s) = \frac{1}{1 + a_1 s + a_2 s^2} = \frac{1}{1 + 1,4142s + s^2} \quad (4.6)$$

Pro výpočet činitelů se využije tab. 4.2 pro převod analogové dolní propusti na pásmovou propust bilineární transformací.

Tab. 4.2 – Bilineární transformace dolní propusti na pásmovou propust

| | | | |
|-----------------|--|--|--|
| Pásmová propust | $s \rightarrow \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1}$ | $\alpha = \frac{4 - \Omega^2 T^2}{4 + \Omega^2 T^2}$ | $\beta = \frac{4 + \Omega^2 T^2}{2\Delta\Omega T}$ |
|-----------------|--|--|--|

4.1.1 Výpočet filtru s kmitočtem f_1

Výpočet činitelů:

$$f_1 = 500 \quad (4.7)$$

$$\alpha = \frac{4 - \Omega^2 T_{vz}^2}{4 + \Omega^2 T_{vz}^2} = \frac{4 - (2\pi f_1)^2 T_{vz}^2}{4 + (2\pi f_1)^2 T_{vz}^2} = \frac{4 - (1000\pi)^2 \times \frac{1}{10000^2}}{4 + (1000\pi)^2 \times \frac{1}{10000^2}} = 0,95184 \quad (4.8)$$

$$\beta = \frac{4 + \Omega^2 T_{vz}^2}{2\Delta\Omega T_{vz}} = \frac{4 + (2\pi f_1)^2 T_{vz}^2}{2\Delta\Omega T_{vz}} = \frac{4 + (1000\pi)^2 \times \frac{1}{10000^2}}{2 \times 24360 \times \frac{1}{10000}} = 0,84128 \quad (4.9)$$

kde α – činitel α

β – činitel β

T_{vz} – vzorkovací perioda, s,

f_1 – zvolený kmitočet, Hz,

$\Delta\Omega$ – šířka pásma

Ω – úhlový kmitočet

Filtrační funkce $F(z)$ v oblasti operátoru z se získá substitucí (4.10) ve tvaru:

$$s \rightarrow \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} \quad (4.10)$$

$$F(z) = \frac{1}{1 + 1,4142\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} + \left(\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1}\right)^2} = \quad (4.11)$$

$$\begin{aligned}
& \frac{1}{1 + 1,4142 \times 084128 \times \frac{z^2 - 1,90368z + 1}{z^2 - 1}} + \\
& \frac{1}{\left(0,84128 \times \frac{z^2 - 1,90368z + 1}{z^2 - 1}\right)^2} = \\
& \frac{z^4 - 2z^2 + 1}{2,89749z^4 - 4,95955z^3 + 1,9804z^2 - 0,42979z + 0,51801} \times \\
& \frac{1}{2,89749z^{-4}} = \\
& \frac{1}{2,89749z^{-4}} = \\
& \frac{0,34513 - 0,69025z^{-2} + 0,34513z^{-4}}{1 - 1,71167z^{-1} + 0,68349z^{-2} - 0,14833z^{-3} + 0,17878z^{-4}} = \frac{Y(z)}{X(z)}
\end{aligned}$$

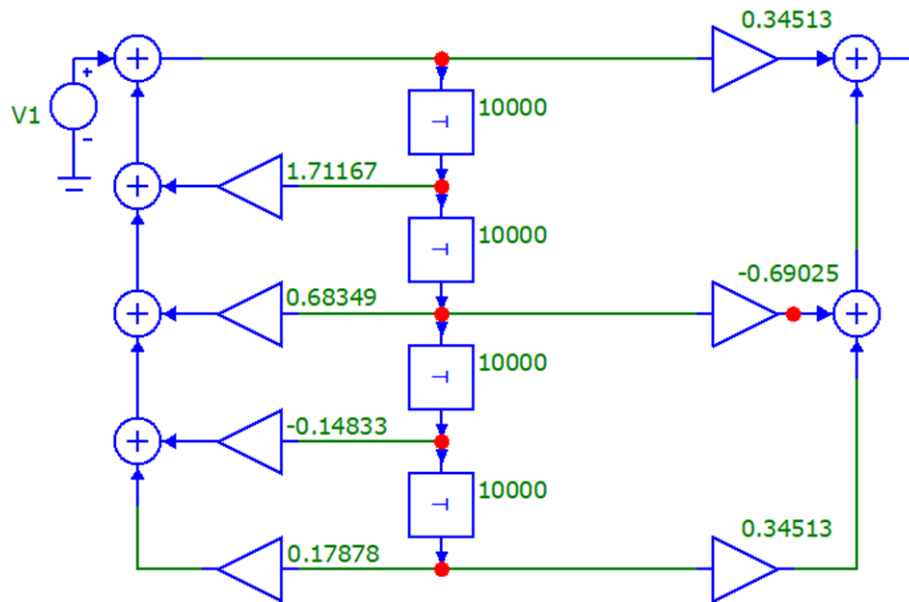
Rovnice se rozšíří pomocnou proměnou $P(z)$:

$$\frac{Y(z)}{P(z)} \times \frac{P(z)}{X(z)} = (0,34513 - 0,69025z^{-2} + 0,34513z^{-4}) \times \tag{4.12}$$

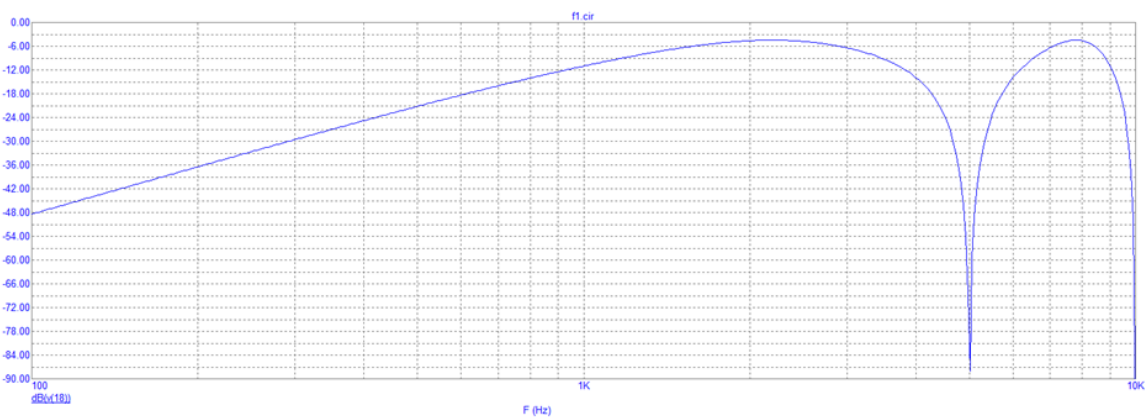
$$\frac{1}{1 - 1,71167z^{-1} + 0,68349z^{-2} - 0,14833z^{-3} + 0,17878z^{-4}}$$

$$Y(z) = 0,34513P(z) - z^{-2}0,69025P(z) + z^{-4}0,34513P(z) \tag{4.13}$$

$$\begin{aligned}
P(z) = X(z) + z^{-1}1,71167P(z) + z^{-2}0,68349P(z) - z^{-3}0,14833P(z) + \\
z^{-4}0,17878P(z)
\end{aligned} \tag{4.14}$$



Obr. 4.1 – Blokové schéma filtru s kmitočtem f_1



Obr. 4.2 – Výsledek simulace pro kmitočtet f_1

4.1.2 Výpočet filtru s kmitočtem f_2

Výpočet činitelů:

$$f_2 = 1000 \quad (4.15)$$

$$\alpha = \frac{4 - \Omega^2 T_{vz}^2}{4 + \Omega^2 T_{vz}^2} = \frac{4 - (2\pi f_2)^2 T_{vz}^2}{4 + (2\pi f_2)^2 T_{vz}^2} = \frac{4 - (2000\pi)^2 \times \frac{1}{10000^2}}{4 + (2000\pi)^2 \times \frac{1}{10000^2}} = 0,82034 \quad (4.16)$$

$$\beta = \frac{4 + \Omega^2 T_{vz}^2}{2\Delta\Omega T_{vz}} = \frac{4 + (2\pi f_2)^2 T_{vz}^2}{2\Delta\Omega T_{vz}} = \frac{4 + (2000\pi)^2 \times \frac{1}{10000^2}}{2 \times 24360 \times \frac{1}{10000}} = 0,90205 \quad (4.17)$$

kde α – činitel α

β – činitel β

T_{vz} – vzorkovací perioda, s,

f_2 – zvolený kmitočet, Hz,

$\Delta\Omega$ – šířka pásma

Ω – úhlový kmitočet

Filtrační funkce $F(z)$ v oblasti operátoru z se získá substitucí (4.16) ve tvaru

$$s \rightarrow \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} \quad (4.18)$$

$$F(z) = \frac{1}{1 + 1,4142 \times \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} + \left(\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{1 + 14142 \times 0,90205 \times \frac{z^2 - 1,64068z + 1}{z^2 - 1} + \left(0,90205 \times \frac{z^2 - 1,64068z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{\left(0,90205 \times \frac{z^2 - 1,64068z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{z^4 - 2z^2 + 1}{3,08937z^4 - 4,76301z^3 + 1,81773z^2 - 0,57705z + 0,53801} \times$$

$$\frac{1}{3,08937z^{-4}} =$$

$$\frac{1}{3,08937z^{-4}} =$$

$$\frac{0,32369 - 0,64738z^{-2} + 0,32369z^{-4}}{1 - 1,54174z^{-1} + 0,58838z^{-2} - 0,18679z^{-3} + 0,17415z^{-4}} = \frac{Y(z)}{X(z)} \quad (4.19)$$

Rovnice se rozšíří pomocnou proměnou $P(z)$

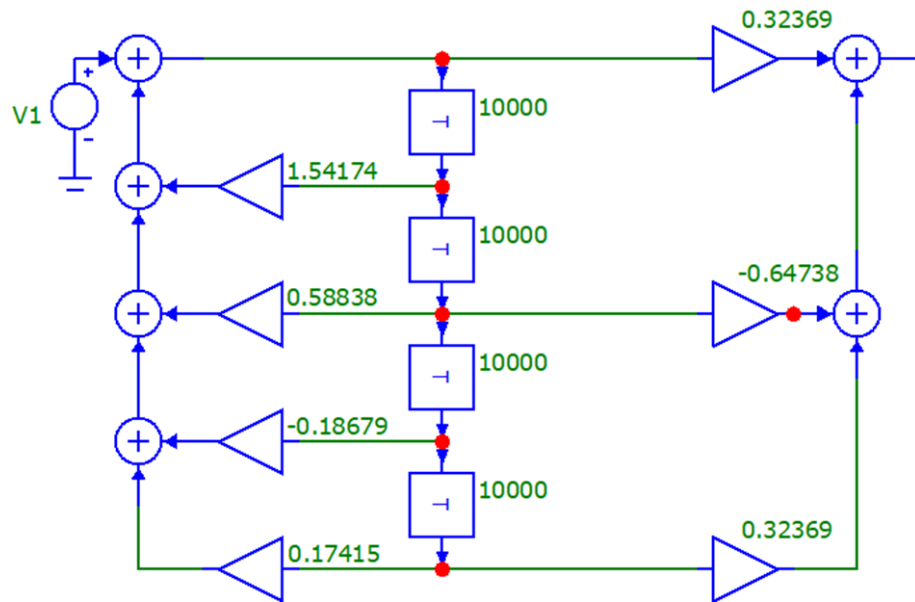
$$\frac{Y(z)}{P(z)} \times \frac{P(z)}{X(z)} = (0,32369 - 0,64738z^{-2} + 0,32369z^{-4}) \times$$

$$\frac{1}{1 - 1,54174z^{-1} + 0,58838z^{-2} - 0,18679z^{-3} + 0,17415z^{-4}} \quad (4.20)$$

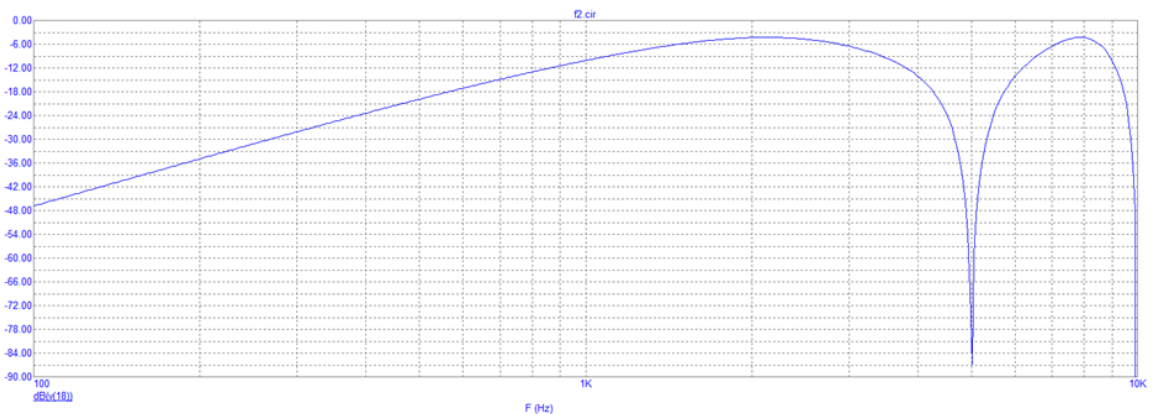
$$Y(z) = 0,32369P(z) - z^{-2}0,64738P(z) + z^{-4}0,32369P(z) \quad (4.21)$$

$$P(z) = X(z) + z^{-1}1,54174P(z) + z^{-2}0,58838P(z) - z^{-3}0,18679P(z) +$$

$$z^{-4}0,17415P(z) \quad (4.22)$$



Obr. 4.4 – Blokové schéma filtru s kmitočtem f_2



Obr. 4.3 – Výsledek simulace pro kmitočet f_2

4.1.3 Výpočet filtru s kmitočtem f_3

Výpočet činitelů

$$f_3 = 2000 \quad (4.23)$$

$$\alpha = \frac{4 - \Omega^2 T_{vz}^2}{4 + \Omega^2 T_{vz}^2} = \frac{4 - (2\pi f_3)^2 T_{vz}^2}{4 + (2\pi f_3)^2 T_{vz}^2} = \frac{4 - (4000\pi)^2 \times \frac{1}{10000^2}}{4 + (4000\pi)^2 \times \frac{1}{10000^2}} = 0,43391 \quad (4.24)$$

$$\beta = \frac{4 + \Omega^2 T_{vz}^2}{2\Delta\Omega T_{vz}} = \frac{4 + (2\pi f_3)^2 T_{vz}^2}{2 \times 24360 \times \frac{1}{10000}} = \frac{4 + (4000\pi)^2 \times \frac{1}{10000^2}}{2 \times 24360 \times \frac{1}{10000}} = 1,14514 \quad (4.25)$$

kde α – činitel α

β – činitel β

T_{vz} – vzorkovací perioda, s,

f_3 – zvolený kmitočet, Hz,

$\Delta\Omega$ – šířka pásma

Ω – úhlový kmitočet

Filtrační funkce $F(z)$ v oblasti operátoru z se získá substitucí (4.20) ve tvaru

$$s \rightarrow \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} \quad (4.26)$$

$$F(z) = \frac{1}{1 + 1,4142\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} + \left(\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{1 + 1,4142 \times 1,14514 \times \frac{z^2 - 0,86782z + 1}{z^2 - 1} + \left(1,14514 \times \frac{z^2 - 0,86782z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{\left(1,14514 \times \frac{z^2 - 0,86782z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{3,93081z^4 - 3,68143z^3 + 1,61029z^2 - 0,87063z + 0,69189} \times$$

$$\frac{1}{3,93081z^{-4}} =$$

$$\frac{1}{3,93081z^{-4}} =$$

$$\frac{0,2544 - 0,5088z^{-2} + 0,2544z^{-4}}{1 - 0,93656z^{-1} + 0,40966z^{-2} - 0,22149z^{-3} + 0,17602z^{-4}} = \frac{Y(z)}{X(z)} \quad (4.27)$$

Rovnice se rozšíří pomocnou proměnou $P(z)$

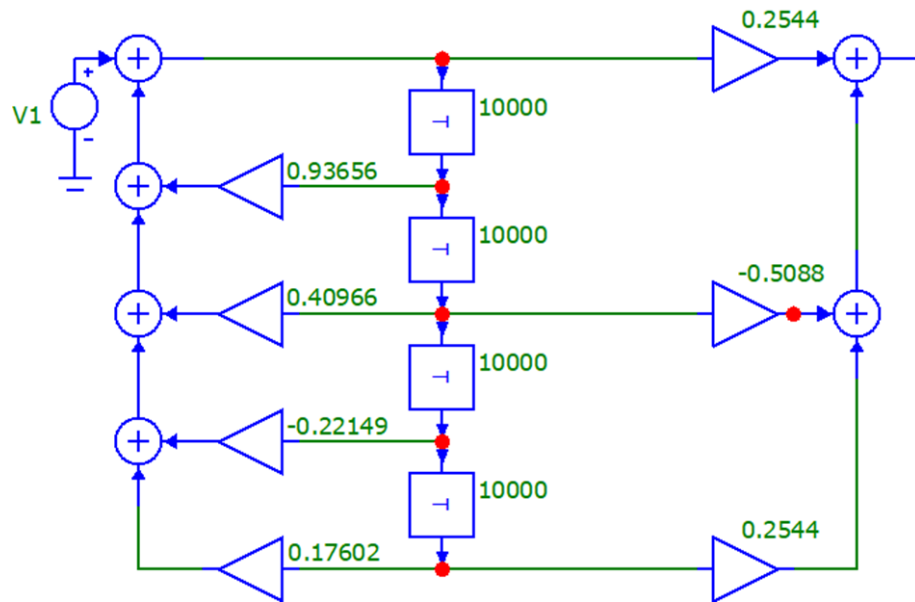
$$\frac{Y(z)}{P(z)} \times \frac{P(z)}{X(z)} = (0,2544 - 0,5088z^{-2} + 0,2544z^{-4}) \times$$

$$\frac{1}{1 - 0,93656z^{-1} + 0,40966z^{-2} - 0,22149z^{-3} + 0,17602z^{-4}} \quad (4.28)$$

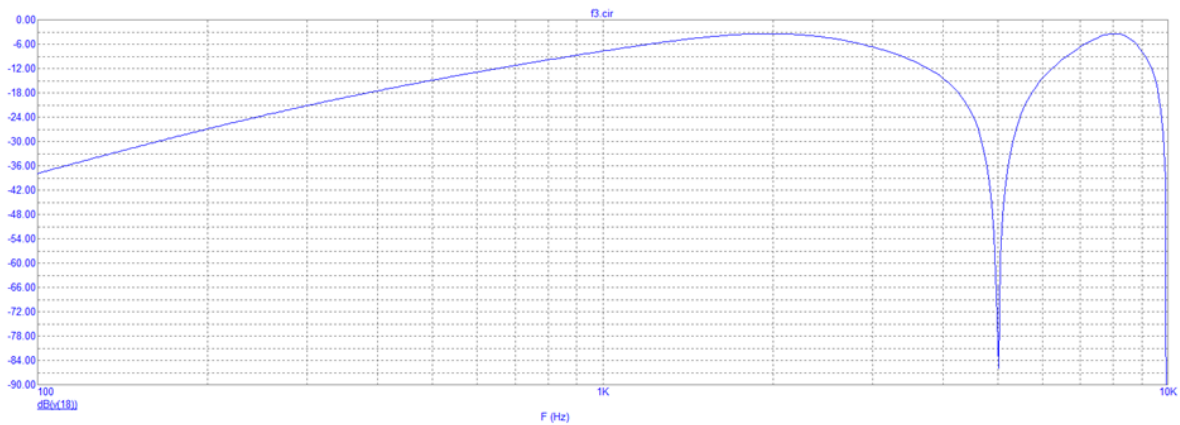
$$Y(z) = 0,2544P(z) - z^{-2}0,5088P(z) + z^{-4}0,2544P(z) \quad (4.29)$$

$$P(z) = X(z) + z^{-1}0,93656P(z) + z^{-2}0,40966P(z) - z^{-3}0,22149P(z) +$$

$$z^{-4}0,17602P(z) \quad (4.30)$$



Obr. 4.5 – Blokové schéma filtru s kmitočtem f_3



Obr. 4.6 – Výsledek simulace pro kmitočet f_3

4.1.4 Výpočet filtru s kmitočtem f_4

Výpočet činitelů

$$f_4 = 3000 \quad (4.31)$$

$$\alpha = \frac{4 - \Omega^2 T^2}{4 + \Omega^2 T^2} = \frac{4 - (2\pi f)^2 T^2}{4 + (2\pi f)^2 T^2} = \frac{4 - (6000\pi)^2 \times \frac{1}{10000^2}}{4 + (6000\pi)^2 \times \frac{1}{10000^2}} = 0,05917 \quad (4.32)$$

$$\beta = \frac{4 + \Omega^2 T^2}{2\Delta\Omega T} = \frac{4 + (2\pi f)^2 T^2}{2 \Delta\Omega T} = \frac{4 + (4000\pi)^2 \times \frac{1}{10000^2}}{2 \times 24360 \times \frac{1}{10000}} = 1,5503 \quad (4.33)$$

kde α – činitel α

β – činitel β

T_{vz} – vzorkovací perioda, s,

f_4 – zvolený kmitočet, Hz,

$\Delta\Omega$ – šířka pásma

Ω – úhlový kmitočet

Filtrační funkce $F(z)$ v oblasti operátoru z se získá substitucí (4.30) ve tvaru

$$s \rightarrow \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} \quad (4.34)$$

$$F(z) = \frac{1}{1 + 1,4142 \times \beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1} + \left(\beta \times \frac{z^2 - 2\alpha z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{1}{1 + 1,4142 \times 1,5503 \times \frac{z^2 - 0,11834z + 1}{z^2 - 1}} +$$

$$\frac{1}{\left(1,5503 \times \frac{z^2 - 0,11834z + 1}{z^2 - 1}\right)^2} =$$

$$\frac{z^4 - 2z^2 + 1}{5,59586z^4 - 0,82829z^3 + 2,84052z^2 - 0,30939z + 1,211} \times$$

$$\frac{1}{5,59586z^{-4}} =$$

$$\frac{1}{5,59586z^{-4}} =$$

$$\frac{0,1787 - 0,3574z^{-2} + 0,1787z^{-4}}{1 - 0,14802z^{-1} + 0,50761z^{-2} - 0,05529z^{-3} + 0,21641z^{-4}} = \frac{Y(z)}{X(z)} \quad (4.35)$$

Rovnice se rozšíří pomocnou proměnou $P(z)$

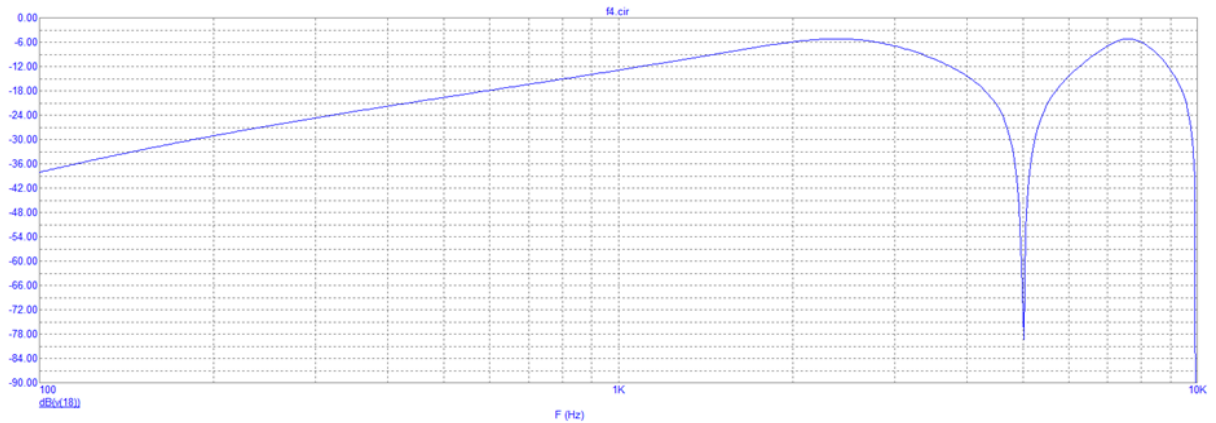
$$\frac{Y(z)}{P(z)} \times \frac{P(z)}{X(z)} = (0,1787 - 0,3574z^{-2} + 0,1787z^{-4}) \times$$

$$\frac{1}{1 - 0,14802z^{-1} + 0,50761z^{-2} - 0,05529z^{-3} + 0,21641z^{-4}} \quad (4.36)$$

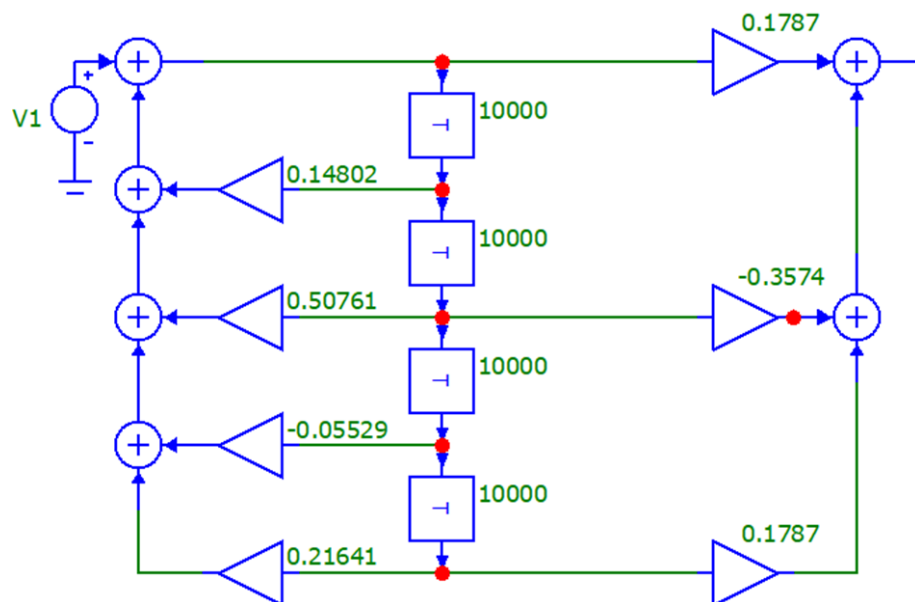
$$Y(z) = 0,1787P(z) - z^{-2}0,3574P(z) + z^{-4}0,1787P(z) \quad (4.37)$$

$$P(z) = X(z) + z^{-1}0,14802P(z) + z^{-2}0,50761P(z) - z^{-3}0,05529P(z) +$$

$$z^{-4}0,21641P(z) \quad (4.38)$$



Obr. 4.8 – Výsledek simulace pro kmitočet f4



Obr. 4.7 – Blokové schéma filtru s kmitočtem f4

Po sestavení blokových schémat byl sestaven vývojový diagram pro program. Realizace byla provedena pomocí tabulek, v kterých byly předem vypočítány hodnoty součinů vzorků s koeficienty. Výpočet při běhu programu potom spočívá pouze v načítání součinů z tabulek. Tímto způsobem je snížen čas výpočtu a obsazení programové i datové paměti. Níže je okomentovaný výpis hlavní části programu pro kmitočet 500 Hz. Na obr. 4.9 je vývojový diagram číslicového filtru, z kterého se dá pochopit princip funkce.

```
// Timer vyvola preruseni kazdych 10ms
ISR(TIMERA1_COMPA_vect)
{
    static uint16_t casovac2 = 0; // casovac 2 pro mereni napeti
    if (!(PIND & 0x10)) {casovac=0; rezim=1; refresh=1; tlacitko=1;}
    // Stisk tlacitka 1
}
```

```

if (!(PIND & 0x20)) {casovac=0; rezim=1; refresh=1; tlacitko=2;}
// Stisk tlacitka 2
if (!(PIND & 0x40)) {casovac=0; rezim=1; refresh=1; tlacitko=3;}
// Stisk tlacitka 3
if (!(PIND & 0x80)) {casovac=0; rezim=1; refresh=1; tlacitko=4;}
// Stisk tlacitka 4
if (casovac < 400) casovac++; // Takt 10ms - 4 sekundy = 400
// Casovac 4 sekund - pokud je to mene tak zvysuj promennou
else
{
    casovac2++; // Casovac 2 - pro mereni napeti
    if (casovac2==10) // Merime 10x za sekundu
    {
        casovac2=0; // Nulujeme pomocny citac casu 100ms
        rezim=0; // Prepiname na rezim 0 - zobrazeni napeti
        refresh=1; // Obnova displeje
    }
}
}
#include "Tab1.txt" //načtení tabulky součinů vzorků s koeficienty pro f500 Hz

//koeficienty pro vypocet p (f500):
#define kp1(n) (int)(pgm_read_word(&k1[n+512]))
#define kp2(n) (int)(pgm_read_word(&k2[n+512]))
#define kp3(n) (int)(pgm_read_word(&k3[n+512]))
#define kp4(n) (int)(pgm_read_word(&k4[n+512]))

//koeficienty pro vypocet y (f500):
#define ky(n) (int)(pgm_read_word(&k5[n+512]))
#define ky2(n) (int)(pgm_read_word(&k6[n+512]))
#define ky4(n) (int)(pgm_read_word(&k5[n+512]))

//promenne pro vypocet:
volatile int x; //vstupni vzorek
volatile int y=0; //vystupni vzorek
volatile int p; //mezivysledek
volatile int p1=0,p2=0,p3=0,p4=0; //pameti
volatile unsigned dataDAC; //upravena hodnota y, 12bitova data
volatile unsigned char cntDAC=0; //citac pro rizeni SPI komunikace

//obsluha preruseni A/D prevodniku:
ISR(ADC_vect)
{
    TIFR0=1<<OCF0A; //nulovani priznaku c/c 0
    sei(); //povoleni vnoreneho preruseni
    //odeslani predochozi vysledku:
    dataDAC=(y+512)<<2; //odebrani znamenka, rozsireni na 12 bitu
    //cteni vstupniho vzorku:
    x=(ADCH<<2)-512; //uprava na cislo se znamenkem
    //vlastni vypocet:
    switch (tlacitko)
    {

```

```

    case 1:
        p=x+kp1(p1)+kp2(p2)+kp3(p3)+kp4(p4);
        y=ky(p)+ky2(p2)+ky4(p4);
    case 2:
        p=x+kp5(p1)+kp6(p2)+kp7(p3)+kp8(p4);
        y=ky5(p)+ky6(p2)+ky7(p4);
    case 3:
        p=x+kp9(p1)+kp10(p2)+kp11(p3)+kp12(p4);
        y=ky8(p)+ky9(p2)+ky10(p4);
    case 4:
        p=x+kp13(p1)+kp14(p2)+kp15(p3)+kp16(p4);
        y=ky11(p)+ky12(p2)+ky13(p4);
}
//posun pameti:
p4=p3;
p3=p2;
p2=p1;
p1=p;
int main(void)
{
    GPIO_init(); // Inicializace portu
    Timer1_init(20000); // 16MHz / 8 / 20000 = 10ms cas
    ADC_init(); // Inicializace AD prevodniku

    lcd_init(LCD_DISP_ON); // Inicializace LCD

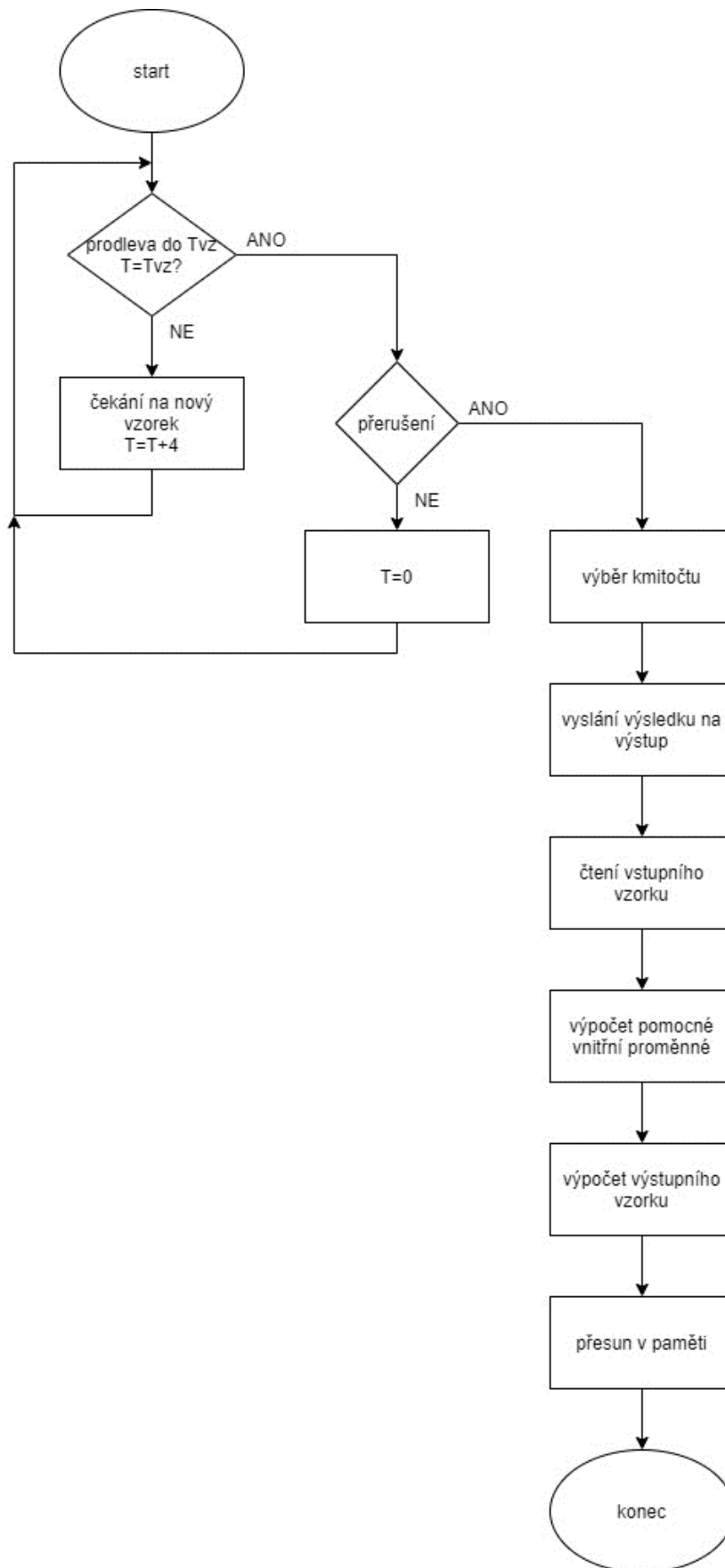
    sei(); // Povoleni preruseni

    {
        //konfigurace c/c 0:
        TCCR0A=WGM_CTC; //rezim CTC
        TCCR0B=F0_8; //preddelicka 8
        OCR0A=OC0APRE; //predvolba pro OCR0A

        //konfigurace A/D prevodniku:
        ADMUX=(1<<REFS1)|(1<<REFS0)|(1<<ADLAR); //vnitni ref., 8bitu
        ADCSRA=(1<<ADEN)|(1<<ADSC)|(1<<ADIF)|(1<<ADIFSC)|(1<<ADIFRSTF);
        //zapnuti, autospousteni, del. 64
        ADCSRB=ADT_CTC0; //spousteni pres CTC0

        //inicializace SPI:
        DDRB=0xFF; //aktivuj výstupy
        SPCR=(1<<SPIE)|(1<<SPE)|(1<<MSTR); //režim SPI
        SPSR=(1<<SPI2X); //rychlost f0/2 (10 MHz)
    }
    while (1)

```



Obr. 4.9 – Vývojový diagram číslicového filtru

5 ZÁVĚR

Použitím číslicového filtru pro selektivní voltmetr se zvýší flexibilita, vlastnosti filtru se dají snadno upravit změnou softwaru, u analogových filtrů musí většinou dojít ke změně návrhu celého obvodu. Přesnost digitálního filtru nezávisí na vnějších podmínkách, ale hlavně na zvoleném rozlišení. Pro tuto práci jsem zvolil rozlišení 10 bitů.

Při konstrukci selektivního voltmetru byla z důvodu jednoduchosti klávesnice pro výběr konkrétní frekvence nahrazena čtyřmi samostatnými tlačítky.

Pomocí programu MicroCap byly vytvořeny grafy filtru podle vypočítaných koeficientů. Ve výsledku simulace jsou vidět zřetelná minima. Zobrazované napětí na LCD displeji bez připojeného generátoru odpovídá mému očekávání. Funkci lze prakticky ověřit připojením signálu konkrétního kmitočtu na vstup navrženého voltmetru a paralelně k němu by se připojil přesný voltmetr. Výsledky voltmetrů se porovnají a tím se zjistí funkčnost.

Projekt by se dal v budoucnu vylepšit rozšířením o grafický LCD displej, případně záměnou za původní alfanumerický LCD displej a úpravou softwaru, na kterém by se vykresloval signál po průchodu číslicovým filtrem. Díky tomu by došlo k rychlejšímu zjištění funkčnosti bez připojování dalších komponentů.

POUŽITÁ LITERATURA

- BRTNÍK, B. 2014. *Základy obvodové techniky 1*. Praha: BEN – technická literatura. 180 s. ISBN 978-80-7300-5.
- DOSTÁL, T.; VRBA, K. 1993. *Elektrické filtry*. Brno: Nakladatelství VUT. 100 s. ISBN 802-14-0496-5.
- BRTNÍK, B.; MATOUŠEK D. 2012. *Digital signal processing algorithms*. Saarbrücken (Německo): LAP LAMBERT Academic Publishing. 108 p. ISBN 978-3-8473-4411-7.
- HÁJEK, K.; SEDLÁČEK J. 2002. *Kmitočtové filtry*. Praha: BEN – technická literatura. 529 s. ISBN 80-7300-023-7.
- Davídek, V.; Sovka P. 1996. *Číslicové zpracování signálů a implementace*. Praha: Nakladatelství ČVUT. 240 s.
- BRTNÍK, B.; MATOUŠEK D. Aplikace procesoru ATmega644 v algoritmech číslicového zpracování sinálu. *ARADIO*, ročník XV (2010), č. 1, s. 3 – 40. ISSN 1211-3557.
- SMÉKAL, Z. 1993. *Číslicové filtry*. Brno: Nakladatelství VUT. 136 s. ISBN 80-214-0500-7.
- BRTNÍK, B.; MATOUŠEK D. Aplikace algoritmů číslicového zpracování signálu 1. Praha: BEN – technická literatura. 174 s. ISBN 978-80-7300-478-1.
- BUMBA, J. 2011. *Programování mikroprocesorů: praktický návod nejen pro mikroprocesory PIC*. Brno: Computer Press. 135 s. ISBN 978-80-251-2838-1.
- JANEČEK, J. 1999. *Projektování mikro počítačových systémů*. [on-line]. Nakladatelství ČVUT. 132 s. [cit. 2020-07-30]. Dostupné na: <http://elektrotechnika.vesele.info/rubriky/vyvoj-pocitacu/vyvoj-pc>.
- Matoušek, D. 2013. *Aplikace mikrokontrolérů ATmega644*. Praha: BEN – technická literatura. 192 s. ISBN 978-80-7300-492-7.
- Základy ovládání mikrokontroléru ATmega(16/32): Dvouřádkový LCD displej*. 2014. [on-line]. tajned.cz. Židlochovice. [cit. 2020-07-28]. Dostupné na: <http://www.tajned.cz/2014/11/zaklady-ovladani-mikrokontroleru-atmega-14-dil-dvouradkovy-lcd-displej/>.
- Wall mount SYS1421-Wxx series* [on-line]. Sunny. 2 s. [cit. 2020-07-29]. Dostupné na: <https://www.tme.eu/Document/11f92fcf90262ceb2306a845c3cae8b9/SYS1421.pdf>.
- VÁŇA, V. 2003. *Mikrokontroléry Atmel AVR: programování v jazyce C*. Praha: BEN – technická literatura. 216 s. ISBN 80-7300-102-0.
- HRBÁČEK, J. 2002. *Komunikace mikrokontroléru s okolím 1*. Praha: BEN – technická literatura. 160 s. ISBN 80-86056-42-2.
- UART*. [on-line]. src.athaj.cz. [cit. 2020-07-27]. Dostupné na: <http://src.athaj.cz/teaching/rev/uart>
- Externí sériové sběrnice SPI a I²C*. 2008. [on-line]. Root.cz. Praha. [cit. 2020-07-31]. Dostupné na: <https://www.root.cz/clanky/externi-seriove-sbernice-spi-a-i2c/>.
- I²C (TWI) – sériová komunikace po dvou vodičích s adresací*. 2014. [on-line]. zavavov.cz. Brno. [cit. 2020-07-28]. Dostupné na: <http://www.zavavov.cz/cz/elektrotechnika/komunikacni-sbernice/68-i2c-twi-seriova-komunikace-po-dvou-vodicich-s-adresaci/>.

PŘÍLOHY

A – CD

B – Uživatelský manuál

Příloha k bakalářské práci
Selektivní voltmetr na bázi číslicového filtru
Josef Jelínek

CD

OBSAH

- 1 Text bakalářské práce ve formátu PDF.
- 2 Zdrojový kód aplikace mikropočítače se všemi použitými knihovnamí.
- 3 Schéma zapojení zařízení v prostředí EAGLE i ProfiCAD, návrh desky plošných spojů.
- 4 Soubor s grafy filtru v prostředí Micro-Cap.

Příloha k bakalářské práci

Selektivní voltmetr na bázi číslicového filtru

Josef Jelínek

**UŽIVATELSKÝ MANUÁL
VÝVOJOVÉ PROSTŘEDÍ**

OBSAH

| | |
|------------------------------|-----|
| SEZNAM ILUSTRACÍ | B-3 |
| SEZNAM TABULEK..... | B-4 |
| 1 ZAPOJENÍ..... | B-5 |
| 2 VÝVOJOVÉ PROSTŘEDÍ..... | B-6 |
| 3 VÝPIS CELÉHO PROGRAMU..... | B-8 |

SEZNAM ILUSTRACÍ

| | |
|--|-----|
| Obr. 1.1 – Rozložení pinů ATmega644 | B-5 |
| Obr. 2.1 – Dialogové okno pro vytvoření projektu..... | B-6 |
| Obr. 2.2 – Vývojové prostředí | B-7 |
| Obr. 2.3 – Dialogové okno pro výběr typu procesoru | B-7 |

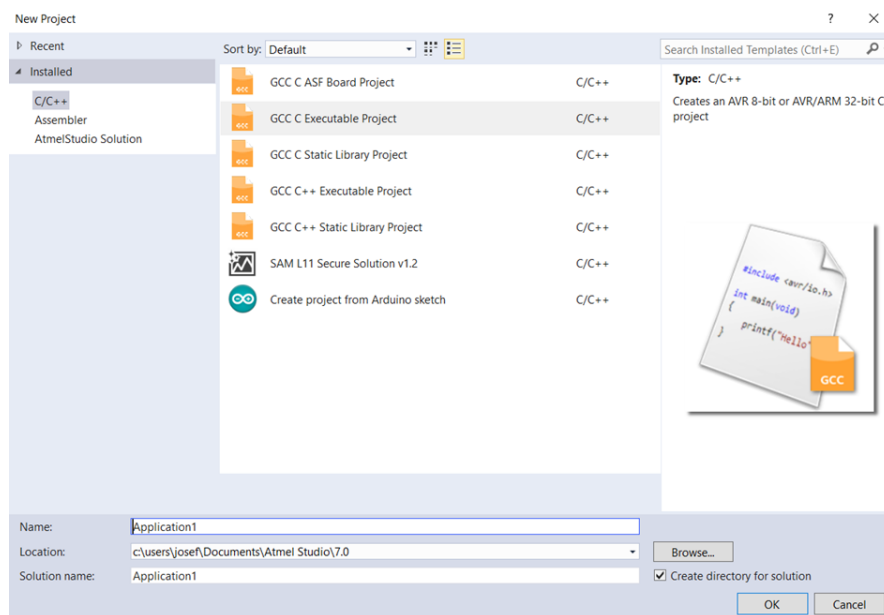
SEZNAM TABULEK

| | |
|--|-----|
| Tab. 1.1 – Zapojení selektivního voltmetru | B-5 |
|--|-----|

2 VÝVOJOVÉ PROSTŘEDÍ

Aplikace pro mikro počítač začíná vytvořením projektu ve vývojovém prostředí. Pro tento projekt bylo použito AVR Studio.

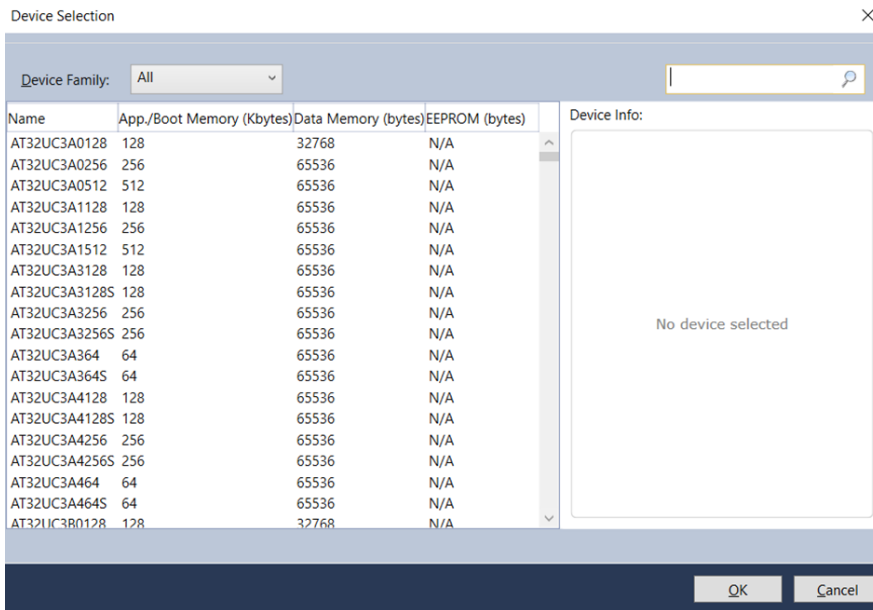
Pokud chceme vytvořit nový projekt, spustíme AVR Studio a stiskneme tlačítko New Project. Potom se zobrazí dialogové okno, viz obr. 2.1. Je třeba zvolit možnost C/C++ (programování v jazyce C), následně vybrat GCC C Executable Project a zadat název projektu a adresář. Pokračujeme stisknutím tlačítka OK.



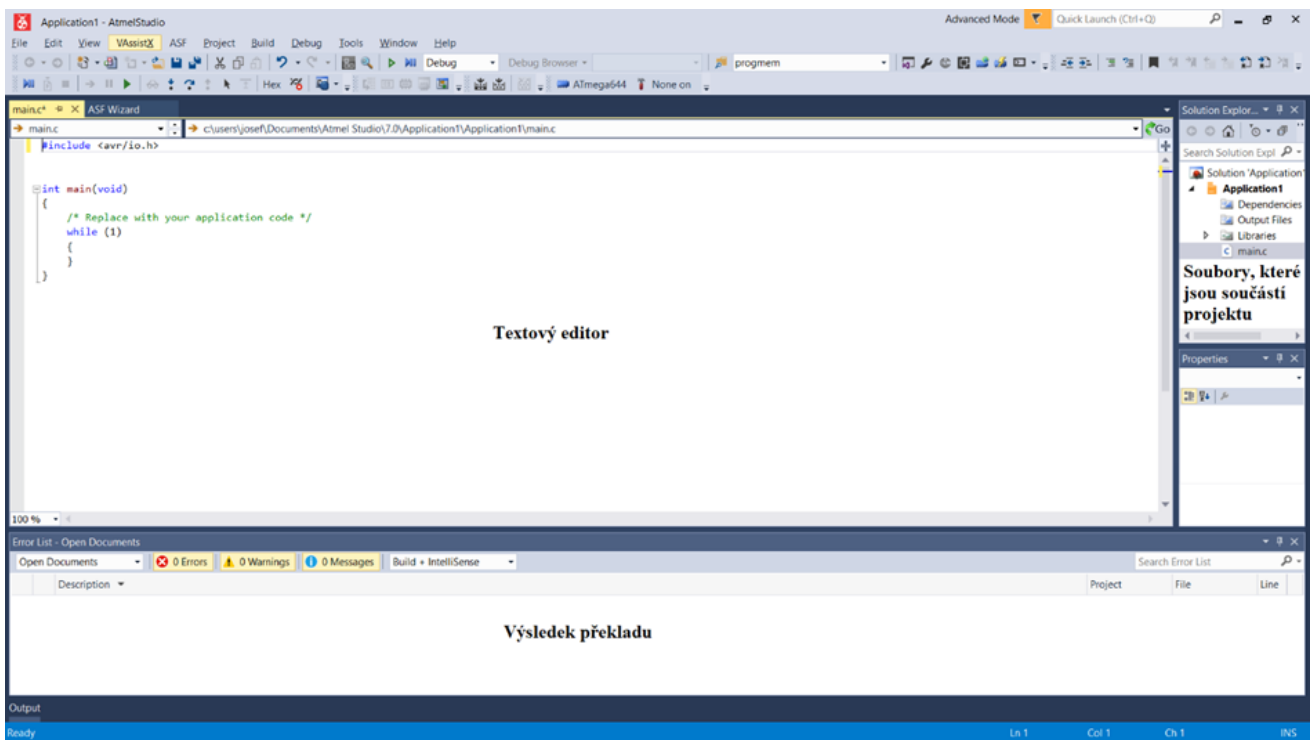
Obr. 2.1 – Dialogové okno pro vytvoření projektu

Následně se zobrazí dialogové okno (obr. 2.2), kde se zvolí použitý procesor (v tomto případě ATmega644). Stiskem tlačítka OK dokončíme vytvoření nového projektu. Po provedení těchto kroků je vše připraveno pro zápis programu. Na obr. 2.3 jsou jednotlivá popsána okna vývojového prostředí.

Po dopsání programu je možnost simulace hotového programu, jako je krokování programu, sledování obsahu registrů a pamětí. Pokud program funguje bez chyb, je možné zkusit ho nahrát do mikro počítače a ověřit jeho činnost na hardwarové úrovni. Pro nahrání do mikro počítače slouží aplikace COMXKIT.



Obr. 2.2 – Dialogové okno pro výběr typu procesoru



Obr. 2.3 – Vývojové prostředí

3 VÝPIS CELÉHO PROGRAMU

```
#define F_CPU 16000000ul
#include <stdio.h>
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
#include <avr/pgmspace.h>
#include "lcd/lcd.h"
#define __AVR_ATmega644__ 1
#define NFP 128 //nasobici faktor pro koeficienty p
#define NFY 128 //nasobici faktor pro koeficienty pri vypoctu y
#define F0_8 2 //predelicka citace/casovace
#define WGM CTC 2 //rezim cinnosti citace/casovace
#define OC0APRE (250-1) //delka citace, odpovida pozadovanemu fvz 10kHz
#define ADP_64 6 //predelicka A/D prevodniku
#define ADT_CTC0 3 //zdroj spousteni A/D prevodniku

char buffer[20]; // Buffer pro displej
volatile uint8_t rezim=0; // 0 - napeti 1 - kmitocet
volatile uint8_t tlacitko=0; // Hodnota stiskleho tlacitka
volatile uint8_t refresh=0; // Priznak obnovy displeje
volatile uint16_t casovac = 400; // Pomocny casovac pro 4 sekundy
uint16_t kmitocet[] = {500,1000,2000,3000}; // Tabulka frekvenci

// vstup A/D prevodniku
void GPIO_init();
void Timer1_init(uint16_t strop);
void ADC_init();
uint16_t ADC_read();

// Timer vyvola preruseni kazdych 10ms
ISR(TIMER1_COMPA_vect)
{
    static uint16_t casovac2 = 0; // casovac 2 pro mereni napeti
    if (!(PIND & 0x10)) {casovac=0; rezim=1; refresh=1; tlacitko=1;}
    // Stisk tlacitka 1
    if (!(PIND & 0x20)) {casovac=0; rezim=1; refresh=1; tlacitko=2;}
    // Stisk tlacitka 2
    if (!(PIND & 0x40)) {casovac=0; rezim=1; refresh=1; tlacitko=3;}
    // Stisk tlacitka 3
    if (!(PIND & 0x80)) {casovac=0; rezim=1; refresh=1; tlacitko=4;}
    // Stisk tlacitka 4
    // Takt 10ms - 4 sekundy = 400
    if (casovac < 400) casovac++; // Casovac 4 sekund - pokud je to mene tak
    zvysuj promennou
    else
    {
        casovac2++; // Casovac 2 - pro mereni napeti
        if (casovac2==10) // Merime 10x za sekundu
        {
            casovac2=0; // Nulujeme pomocny citac casu 100ms
            rezim=0; // Prepiname na rezim 0 - zobrazeni napeti
            refresh=1; // Obnova displeje
        }
    }
}

#include "Tab1.txt" //načtení tabulky součinnů vzorků s koeficienty pro f500 Hz
#include "Tab2.txt" //načtení tabulky součinnů vzorků s koeficienty pro f1000 Hz
#include "Tab3.txt" //načtení tabulky součinnů vzorků s koeficienty pro f2000 Hz
#include "Tab4.txt" //načtení tabulky součinnů vzorků s koeficienty pro f3000 Hz
```

```

//koeficienty pro vypocet p (f500):
#define kp1(n) (int)(pgm_read_word(&k1[n+512]))
#define kp2(n) (int)(pgm_read_word(&k2[n+512]))
#define kp3(n) (int)(pgm_read_word(&k3[n+512]))
#define kp4(n) (int)(pgm_read_word(&k4[n+512]))

//koeficienty pro vypocet y (f500):
#define ky(n) (int)(pgm_read_word(&k5[n+512]))
#define ky2(n) (int)(pgm_read_word(&k6[n+512]))
#define ky4(n) (int)(pgm_read_word(&k5[n+512]))

//koeficienty pro vypocet p (f1000):
#define kp5(n) (int)(pgm_read_word(&ko1[n+512]))
#define kp6(n) (int)(pgm_read_word(&ko2[n+512]))
#define kp7(n) (int)(pgm_read_word(&ko3[n+512]))
#define kp8(n) (int)(pgm_read_word(&ko4[n+512]))

//koeficienty pro vypocet y (f1000):
#define ky5(n) (int)(pgm_read_word(&ko5[n+512]))
#define ky6(n) (int)(pgm_read_word(&ko6[n+512]))
#define ky7(n) (int)(pgm_read_word(&ko5[n+512]))

//koeficienty pro vypocet p (f2000):
#define kp9(n) (int)(pgm_read_word(&koe1[n+512]))
#define kp10(n) (int)(pgm_read_word(&koe2[n+512]))
#define kp11(n) (int)(pgm_read_word(&koe3[n+512]))
#define kp12(n) (int)(pgm_read_word(&koe4[n+512]))

//koeficienty pro vypocet y (f2000):
#define ky8(n) (int)(pgm_read_word(&koe5[n+512]))
#define ky9(n) (int)(pgm_read_word(&koe6[n+512]))
#define ky10(n) (int)(pgm_read_word(&koe5[n+512]))

//koeficienty pro vypocet p (f3000):
#define kp13(n) (int)(pgm_read_word(&koef1[n+512]))
#define kp14(n) (int)(pgm_read_word(&koef2[n+512]))
#define kp15(n) (int)(pgm_read_word(&koef3[n+512]))
#define kp16(n) (int)(pgm_read_word(&koef4[n+512]))

//koeficienty pro vypocet y (f3000):
#define ky11(n) (int)(pgm_read_word(&koef5[n+512]))
#define ky12(n) (int)(pgm_read_word(&koef6[n+512]))
#define ky13(n) (int)(pgm_read_word(&koef5[n+512]))

//promenne pro vypocet:
volatile int x; //vstupni vzorek
volatile int y=0; //vystupni vzorek
volatile int p; //mezivysledek
volatile int p1=0,p2=0,p3=0,p4=0; //pameti
volatile unsigned dataDAC; //upravena hodnota y, 12bitova data
volatile unsigned char cntDAC=0; //citac pro rizeni SPI komunikace

//obsluha preruseni A/D prevodniku:
ISR(ADC_vect)
{
    TIFR0=1<<OCF0A; //nulovani priznaku c/c 0
    sei(); //povoleni vnoreneho preruseni
    //odeslani predochozi vysledku:
    dataDAC=(y+512)<<2; //odebrani znamenska, rozsireni na 12 bitu
    //cteni vstupniho vzorku:
    x=(ADCH<<2)-512; //uprava na cislo se znamenkem
}

```

```

//vlastni vypocet:
switch (tlacitko)
{
    case 1:
        p=x+kp1(p1)+kp2(p2)+kp3(p3)+kp4(p4);
        y=ky(p)+ky2(p2)+ky4(p4);
    case 2:
        p=x+kp5(p1)+kp6(p2)+kp7(p3)+kp8(p4);
        y=ky5(p)+ky6(p2)+ky7(p4);
    case 3:
        p=x+kp9(p1)+kp10(p2)+kp11(p3)+kp12(p4);
        y=ky8(p)+ky9(p2)+ky10(p4);
    case 4:
        p=x+kp13(p1)+kp14(p2)+kp15(p3)+kp16(p4);
        y=ky11(p)+ky12(p2)+ky13(p4);
}
//posun pameti:
p4=p3;
p3=p2;
p2=p1;
p1=p;
}

int main(void)
{
    GPIO_init(); // Inicializace portu
    Timer1_init(20000); // 16MHz / 8 / 20000 = 10ms cas
    ADC_init(); // Inicializace AD prevodniku

    lcd_init(LCD_DISP_ON); // Inicializace LCD

    sei(); // Povoleni preruseni

    {
        //konfigurace c/c 0:
        TCCR0A=WGM_CTC; //rezim CTC
        TCCR0B=F0_8; //preddelicka 8
        OCR0A=OC0APRE; //predvolba pro OCR0A

        //konfigurace A/D prevodniku:
        ADMUX=(1<<REFS1)|(1<<REFS0)|(1<<ADLAR); //vnitri ref., 8bitu
        ADCSRA=(1<<ADEN)|(1<<ADSC)|(1<<ADIF)|(1<<ADIFSC); //zapnuti,
        autospousteni, del. 64
        ADCSRB=ADT_CTC0; //spousteni pres CTC0

        //inicializace SPI:
        DDRB=0xFF; //aktivuj výstupy
        SPCR=(1<<SPIE)|(1<<SPE)|(1<<MSTR); //režim SPI
        SPSR=(1<<SPI2X); //rychlost f0/2 (10 MHz)

        while (1)
        {
            if (refresh==1) // Pokud je priznak obnovy displeje
            {
                refresh=0; // Smazani vlajku obnovy displeje
                if (rezim==0) // Pokud je rezim 0 - merime napeti
                {
                    uint16_t analog = ADC_read(); // Preceteme
                    AD prevodnik
                    uint16_t milivolty = (uint16_t)(uint32_t)(analog * 1000ul
/ 1023 * 5); // Prevod na citelnou variantu
                }
            }
        }
    }
}

```

```

        sprintf(buffer, "%u.%03uV", milivolty/1000, milivolty%1000);
// Vypiseme citelnou variantu
        lcd_gotoxy(0,0);           // Vypis na displej - souradnice
        lcd_puts("Napeti ");      // Vypis na displej - text
        lcd_gotoxy(0,1);         // Vypis na displej - souradnice
        lcd_puts(buffer);         // Vypis na displej - text
    }
    else
    {
        // Jinak vypisujeme udaj o frekvenci dle stisku tlacitka
        sprintf(buffer, "%4uHz", kmitocet[tlacitko-1]);
        lcd_gotoxy(0,0);         // Vypis na displej - souradnice
        lcd_puts("Kmitocet");     // Vypis na displej - text
        lcd_gotoxy(0,1);         // Vypis na displej - souradnice
        lcd_puts(buffer);         // Vypis na displej - text
    }
}
}
}

void GPIO_init()
{
    // Tlacitka - pullup na vstupy
    PORTD |= (1<<PD7)|(1<<PD6)|(1<<PD5)|(1<<PD4);
}

void Timer1_init(uint16_t strop)
{
    TCCR1B |= (1<<WGM12)|(1<<CS11); // Timer 1 CTC mode + CLK/8 = 2 MHz
    TIMSK1 |= (1<<OCIE1A);         // Interrupr enable channel A
    OCR1A = strop;                 // CTC to register - strop timeru
}

void ADC_init()
{
    DIDR0 |= (1<<ADC0D);           // Disable digital input
    ADMUX |= (1<<REFS0);           // Reference to AVCC
    ADCSRA |= (1<<ADEN)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0); // enable + prescaler /
128
}

uint16_t ADC_read()
{
    ADCSRA |= (1<<ADSC);           // Start konverze
    while(ADCSRA & (1<<ADSC));    // Cekame nez konverze probehne
    return ADC;                   // vratime vysledek
}

```