

1 ČÍSLICOVÉ FILTRE A FFT (OPAKOVANIE)

Ciele cvičenia:

- zopakovanie základných pojmov (FIR, IIR filter, bikvad, stabilita ...)
- základný návrh FIR a IIR filtrov v prostredí Matlab
- návrh praktických FIR a IIR filtrov pomocou programu FilterExpress
- výpočet FFT a vizualizácia v prostredí Matlab

1.1 ÚVOD

Číslivé filtre a efektívne algoritmy výpočtu **DFT** (Discrete Fourier Transform), ktoré využívajú rýchle algoritmy výpočtu založené na algoritme **FFT** (Fast Fourier Transform) patria medzi základné algoritmy číslivého spracovania signálov (ČSS). Všetky prakticky využívané číslivé signálové procesory (**DSP** - Digital Signal Processor) majú architektúru optimalizovanú minimálne práve pre tieto dva základné typy algoritmov ČSS. Pomocou týchto algoritmov je možné demonštrovať prakticky všetky podstatné vlastnosti jadier klasických DSP a tiež podstatnú časť vlastností moderných DSP. V rámci cvičení budú tieto vlastnosti demonštrované na moderných procesoroch firmy **Analog Devices ADSP Blackfin** s 16-bitovou aritmetikou v pevnej rádovej čiarku [1],[2]. V predchádzajúcich rokoch boli v rámci cvičení používané staršie procesory Analog Devices ADSP 218x [3].

Vzhľadom na to, že problematika číslivých filtrov a FFT je podrobne preberaná na KEMT FEI TU v Košiciach v iných špecializovaných predmetoch, cieľom tohto cvičenia je zopakovanie základných pojmov z tejto oblasti a opísanie **niektorých praktických metód návrhu** číslivých filtrov v prostredí Matlab (v ďalších častiach budú opisované príkazy Matlabu v. 4.21c s nainštalovaným Signal Processing Toolboxom v. 3.0b a špecializovaného voľne dostupného programu **FilterExpress** od firmy Systolix [4]. Aj keď program [4] je voľne dostupný, umožňuje navrhovať a analyzovať aj pomerne zložité praktické číslivé filtre. V prípade extrémnych nárokov na parametre navrhovaných filtrov resp. ich analýzu je možné použiť komerčne dostupné balíky – napr. [5] (tento toolbox Matlabu však nie je na KEMT dostupný).

1.2 ČÍSLICOVÉ FILTRE

1.2.1 FILTRE S KONEČNOU IMPULZOVOU ODPOVEĎOV

Filtre s konečnou impulzovou odpoveďou (FIR- Finite Impulse Response) patria medzi najjednoduchšie číslivé filtre ako z teoretického, tak aj z realizačného hľadiska. FIR filter N -tého rádu je možné opísať **prenosovou funkciou**

$$H_{FIR}(z) = \sum_{k=0}^N h[k]z^{-k} \quad (1.1)$$

pričom $h[k]$ sú **koeficienty** FIR filtra. **Frekvenčná charakteristika** $H^f(\theta)$ FIR filtra (**komplexná funkcia** reálnej premennej θ) je určená vzťahom

$$H^f(\theta) = H_{FIR}(z) \Big|_{z=e^{j\theta}} \quad (1.2)$$

pričom $\theta \in \langle -\pi, \pi \rangle$ (alebo $\theta \in \langle 0, 2\pi \rangle$) je normalizovaná frekvencia aktuálnej frekvencie F (napr. v Hz, kHz a pod.)

$$\theta = 2\pi \frac{F}{F_{samp}} \quad (1.3)$$

a F_{samp} je použitá **frekvencia vzorkovania**.

Keďže frekvenčná charakteristika je komplexnou funkciou, v praxi sa využívajú amplitúdové a fázové frekvenčné charakteristiky, ktoré sú reálnymi funkciami reálnej premennej θ . V prostredí Matlab je možné zobrazit' obidve charakteristiky pomocou príkazu *freqz*.

Príklad

Zobrazte v prostredí Matlab amplitúdovú a fázovú frekvenčnú charakteristiku FIR filtra s koeficientmi $h_i = 0.1$, $i = 0, 1, \dots, 9$.

Riešenie

Príklad vyriešime v prostredí Matlab zapísaním nasledujúcich príkazov

```
h=[0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1];
freqz(h);
```

Medzi charakteristické vlastnosti FIR filtrov patrí ich **stabilita** (t.j. kvantovaním koeficientov sa FIR filter nemôže stať nestabilným) a možnosť realizácie FIR **filtra s lineárnou fázovo-frekvenčnou charakteristikou**. Nutnou a postačujúcou podmienkou aby FIR filter mal lineárnu fázovú frekvenčnú charakteristiku je symetria jeho koeficientov, ktorú je možné vyjadriť v tvare

$$h[k] = h[N - k], \quad k = 0, 1, \dots, N \quad (1.4)$$

1.2.2 FILTRE S NEKONEČNOU IMPULZOVOU ODPOVEĎOV

Pre **filtre s nekonečnou impulzovou odpoveďou** (IIR – Infinite Impulse Response) je charakteristická prítomnosť spätnej väzby¹, čo sa v prípade prenosovej funkcie prejaví existenciou menovateľa prenosovej funkcie $H_{IIR}(z)$, ktorý je na rozdiel

¹ Existuje však aj možnosť realizácie FIR filtrov pomocou štruktúr so spätnou väzbou. Typickým príkladom je realizácia tzv. hrebeňového (comb) FIR filtra v tvare $y(n) = y(n-1) + x(n) - x(n-N)$, ktorý za predpokladu vhodných počiatočných podmienok realizuje FIR filter $y(n) = x(n) + x(n-1) + \dots + x(n-N+1)$.

od $H_{FIR}(z)$ rôznych od 1 (pre jednoduchosť uvažujme, že rád čitateľa a menovateľa sú zhodné a rovné N):

$$H_{IIR}(z) = \frac{b_0 + b_1 z^{-1} + \dots + b_N z^{-N}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} \quad (1.5)$$

Implementáciu IIR filtra opísaného prenosovou funkciou (1.5) je možné realizovať viacerými spôsobmi. Najznámejšie a v praxi najčastejšie využívané sú realizácie založené na rozklade (1.5) do sekcií druhého rádu, tzv. **bikvadov** podľa vzťahu² (za predpokladu, že N je párne)

$$H_{IIR}(z) = \prod_{k=1}^{N/2} \left(\frac{b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \right) = \prod_{k=1}^{N/2} H_k(z) \quad (1.6)$$

pričom prenosová funkcia k -tého bikvadu $H_k(z)$ je určená vo všeobecnom prípade piatimi koeficientmi $b_{0k}, b_{1k}, b_{2k}, a_{1k}, a_{2k}$. IIR filtre sú charakteristické predovšetkým podmienenou stabilitou, pričom po kvantovaní koeficientov sa IIR filter môže stať nestabilným. Jedným z hlavných dôvodov rozkladu IIR filtra na kaskádne zapojenie bikvadov je predovšetkým podstatne nižšia citlivosť kaskády bikvadov na kvantovanie koeficientov IIR filtra. Navyše, v prípade bikvadov je možné jednoznačne definovať oblasť v ktorej sa musia nachádzať koeficienty bikvadov tak, aby boli všetky bikvady (a tým aj celý IIR filter) stabilné(y).

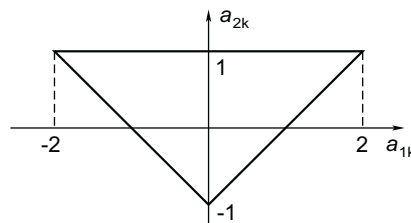
Nutnou podmienkou stability bikvadu je, aby koeficienty menovateľa $H_k(z)$ spĺňali podmienku

$$|a_{jk}| < 2.0 \quad j=1,2 \quad k=1,2,\dots,N/2 \quad (1.7)$$

Postačujúcou podmienkou stability je splnenie vzťahov

$$-1 < a_{2k} < 1, \quad 1 + a_{1k} + a_{2k} > 0, \quad 1 - a_{1k} + a_{2k} > 0 \quad (1.8)$$

ktoré definujú tzv. trojuholník stability zobrazený na Obr.1.



Obr.1 Trojuholník stability bikvadu, body vo vnútri trojuholníka zodpovedajú stabilnému bikvadu

² Niektoré literárne pramene resp. programové balíky používajú pre koeficienty a_k záporné znamienko. Pri praktickom návrhu IIR filtra je preto potrebné správne určenie znamienka, pretože jeho nesprávnym určením sa IIR filter zvyčajne stane nestabilným.

Z pohľadu implementácie pomocou DSP s pevnou rádovou čiarkou je hodnota 2.0 vo vzťahu (1.7) určitým problémom³, pretože interval efektívne využiteľných čísel je u typických DSP obmedzený na interval $\langle -1, 1 \rangle$. V prípade koeficientov a_i $i=1, 2, \dots, N$ vo vzťahu (1.5) nie je možné sformulovať na kontrolu stability podmienky, ktoré by boli také jednoduché ako podmienka (1.7).

Na implementáciu sekcie druhého rádu je možné využiť niekoľko rôznych realizácií. Najznámejšia je **priama forma I**, ktorú je možné opísať diferencnou rovnicou (index k je pre jednoduchosť vynechaný)

$$y(n) = \sum_{j=0}^2 b_j x(n-j) - \sum_{j=1}^2 a_j y(n-j) \quad (1.9)$$

Pri technickej realizácii sú často využívané realizácie (tzv. **kanonické formy**), ktoré využívajú minimálny počet stavebných prvkov (napr. oneskorovacích členov, násobičiek, ...). Medzi najčastejšie využívané patria **kanonická forma I** (tzv. **transponovaná priama forma**) opísaná diferencnou rovnicou

$$y(n) = b_0 x(n) + w_1(n-1) \quad (1.10)$$

$$w_1(n) = b_1 x(n) - a_1 y(n) + w_2(n-1) \quad (1.11)$$

$$w_2(n) = b_2 x(n) - a_2 y(n) \quad (1.12)$$

a **kanonická forma II** (tzv. **priama forma II**) opísaná diferencnou rovnicou

$$w(n) = x(n) - a_1 w(n-1) - a_2 w(n-2) \quad (1.13)$$

$$y(n) = \sum_{j=0}^2 b_j w(n-j) \quad (1.14)$$

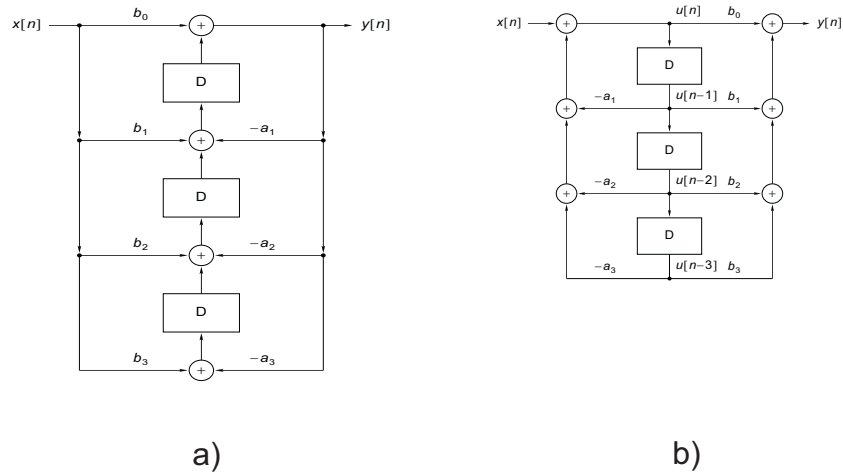
Výhodnou vlastnosťou IIR filtrov v porovnaní s FIR filtrami je menší počet koeficientov, ktoré sú potrebné na realizáciu filtra s určitými vlastnosťami (napr. požadovanou strmosťou medzi priepustným a nepriepustným pásmom).

³ Štandardné DSP tento problém typicky riešia s využitím špecializovaných módov aritmetickej jednotky. V ďalších cvičeniach bude riešenie tohto problému opísané pre ADSP Blackfin od firmy Analog Devices.

Príklad

Na základe rovníc (1.10) - (1.14) nakreslite realizáciu kanonických foriem I a II pre IIR filter rádu $N = 3$ (realizáciu bikvadu dostaneme pre $N = 2$).

Riešenie



Obr.2 Kanonické formy IIR filtra – a) forma I, b) forma II

Príklad

Na základe rovnice (1.9) nakreslite realizáciu priamej formy⁴.

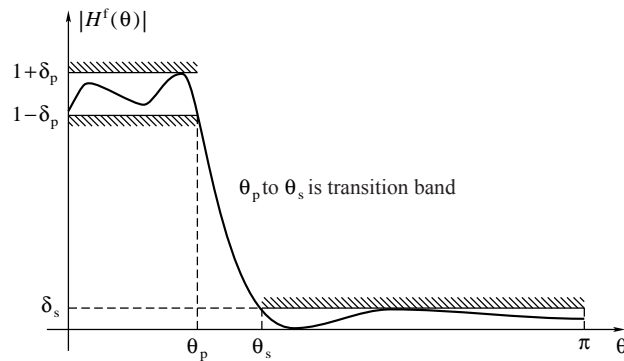
1.2.3 NÁVRH ČÍSLICOVÝCH FILTROV

V procese návrhu je potrebné určiť predovšetkým **koeficienty** filtra, prípadne aj **rád** filtra. Existujú rôzne **návrhové algoritmy**, ktoré sa líšia **kritériom a metódami optimalizácie**. Tieto algoritmy sú preberané v iných špecializovaných predmetoch a preto sa ich opisom nebudeme podrobnejšie zaoberať. Sústreďíme sa predovšetkým na praktické zvládnutie niektorých vybraných návrhových algoritmov s cieľom navrhnuť číslicový filter (typu FIR aj IIR) s požadovanými parametrami.

Medzi najčastejšie požiadavky⁵ v prípade návrhu číslicového filtra patrí **tolerančná schéma**, ktorá je v prípade DP filtra (dolný priepust) naznačená na Obr.3, pričom δ_p je zvlnenie v priepustnom, δ_s je zvlnenie v nepriepustnom pásme a θ_p, θ_s sú normované frekvencie priepustného resp. nepriepustného pásma. Podobne je možné zakresliť tolerančné schémy pre HP, PZ a PP.

⁴ Aj keď uvedená priama forma vyžaduje viac stavebných blokov ako kanonické formy, je často využívaná pri praktickej implementácii práve pomocou DSP.

⁵ V praxi sa samozrejme číslicové filtre navrhujú aj podľa iných špecifikácií.



Obr.3 Tolerančná schéma DP filtra

1.2.3.1 NÁVRH ČÍSLICOVÝCH FILTROV V PROSTREDÍ MATLAB

Návrh číslicových filtrov FIR a IIR v prostredí Matlab budeme demonštrovať na príkladoch. Podrobnejší opis všetkých príkazov Matlab-u je možné získať príkazom

help prikaz

Príklad

Navrhnite DP FIR s pásmom priepustnosti do 3000 Hz, pásmom tlmenia od 5000 Hz, a maximálnym zvlnením v priepustnom a nepriepustnom pásme 0.01 ($20 \log_{10}(1.01) \approx 0.09$ dB) resp. 0.001 ($20 \log_{10}(0.001) = -60$ dB). Pre návrh využite program Matlab, pričom frekvencia vzorkovania je 48000 Hz.

Riešenie

Pre návrh filtrov PP, PZ, DP, HP klasickou metódou oknových funkcií je možné využiť príkaz **fir1**, v prípade požiadavky na zložitejšie priebehy amplitúdovej frekvenčnej charakteristiky je možné využiť príkaz **fir2**. Pri návrhu je potrebné zvoliť typ použitého okna. Metóda oknových funkcií však nie je optimálna (z hľadiska zložitosti navrhnutého filtra, t.j. počtu koeficientov⁶ FIR filtra, ktoré sú potrebné na splnenie požiadaviek zadania) a v praxi je často výhodnejšie využiť zložitejšie návrhové algoritmy (hlavne pokiaľ ich navrhuje počítač), ktoré umožňujú navrhnúť FIR filter s menším počtom koeficientov. V ďalšej časti využijeme tzv. **Remezov výmenný algoritmus**, ktorý patrí do kategórie pomerne výkonných návrhových algoritmov. Najskôr určíme rád FIR filtra, ktorý by mal⁷ splniť zadané požiadavky:

```
[n,fo,mo,w] = remezord( [3000 5000], [1 0], [0.01 0.001], 48000 );
```

a potom učíme koeficienty navrhnutého FIR filtra:

```
h = remez(n,fo,mo,w);
```

navrhnutý FIR filter má

⁶ Počet koeficientov je často kritický predovšetkým pri realizácii filtrov v reálnom čase a preto sa snažíme minimalizovať počet koeficientov navrhovaného FIR filtra.

⁷ Splnenie požiadaviek je potrebné spätne overiť a v prípade ich nespĺnenia zvýšiť rád filtra.

size(h)

koeficientov a má symetrické koeficienty podľa vzťahu (1.4) (je to teda FIR filter s lineárnou fázovo-frekvenčnou charakteristikou). Na spracovanie jednej vstupnej vzorky navrhnutý FIR filter vyžaduje $h=62$ tzv. MAC⁸ operácií. Koeficienty navrhnutého filtra je možné zobrazit príkazom

plot(h)

a amplitúdovú a fázovú frekvenčnú charakteristiku príkazom

freqz(h)

Príklad

Navrhnite DP IIR z predchádzajúceho príkladu a porovnajte zložitosť navrhnutého filtra s FIR filtrom navrhnutým v predchádzajúcom príklade.

Riešenie

Pre návrh IIR je možné opäť využiť niekoľko rôznych návrhových algoritmov. V ďalšej časti vyžijeme návrh tzv. **eliptických filtrov** (niekedy tiež nazývaných Caerove). Tieto filtre využívajú Čebyševovskú aproximáciu v priepustnom aj nepriepustnom pásme a pri návrhu vyžadujú pomerne zložité výpočty (Jakobiho eliptické funkcie, eliptické integrály). S využitím Matlabu je však ich návrh pomerne jednoduchý. Najskôr určíme rád filtra:

```
Rp=20*log10(1.01); % prepočet na decibely
Rs=-20*log10(0.001); % prepočet na decibely
[n,w]=ellipord(3000/(48000/2), 5000/(48000/2), Rp, Rs);
```

a teraz určíme koeficienty filtra

```
[b,a]=ellip(n,Rp,Rs,w);
```

a jeho amplitúdové a frekvenčné charakteristiky

```
freqz(b,a)
```

Na spracovanie jednej vzorky navrhnutý IIR filter vyžaduje len 13 MAC operácií. Ak potrebujeme zobrazit koeficienty na viac desatinných miest ako 4, použijeme príkaz

```
format long
b
a
```

Pri rozklade na sekcie druhého rádu je potrebné zistiť korene polynómov, ktoré je možné nájsť pomocou príkazu **roots** a využiť základné znalosti z algebry o rozklade polynómov.

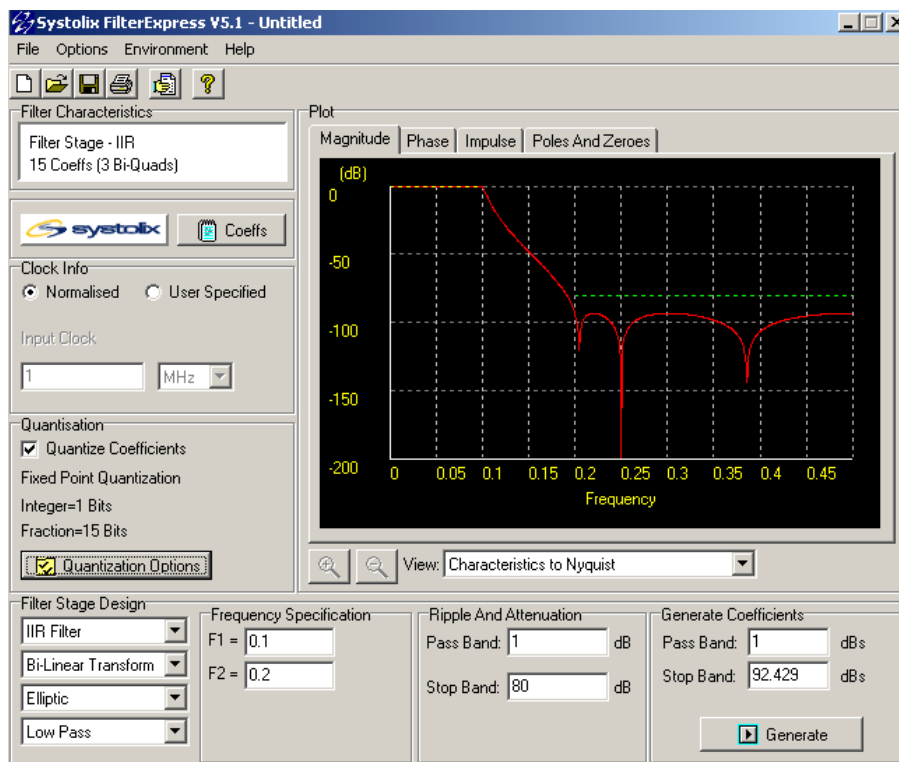
Príklad

Rozložte navrhnutý filter na sekcie druhého rádu.

⁸ MAC operácia (**M**ultiply **A**nd **a**cumulate) je operácia typu $A=A+X*Y$. Operácia MAC je základnou operáciou, pre ktorú sú všetky DSP optimalizované.

1.2.3.2 NÁVRH ČÍSLICOVÝCH FILTROV V PROSTREDÍ FILTEREXPRESS

Inou možnosťou (podstatne pohodlnejšou) ako navrhnuť číslicové filtra je využitie špecializovaných návrhových programov, ktoré umožňujú navrhnuť číslicové filtre dokonca bez znalosti akýchkoľvek príkazov. Umožňujú tiež rozklad navrhnutých IIR filtrov na sekcie druhého rádu. Na počítačoch v učebni je dostupný program Systolix FilterExpress [4]⁹ zobrazený na Obr.4, ktorý umožňuje pohodlným spôsobom navrhnuť prakticky využiteľné FIR a IIR filtre, realizáciu rozkladu IIR filtrov na sekcie druhého rádu a kvantovanie koeficientov filtra. Ich ovládanie je intuitívne a nevyžaduje podrobnejší opis. Podrobnejšie informácie je možné získať z menu help.



Obr.4 Grafické rozhranie programu FilterExpress

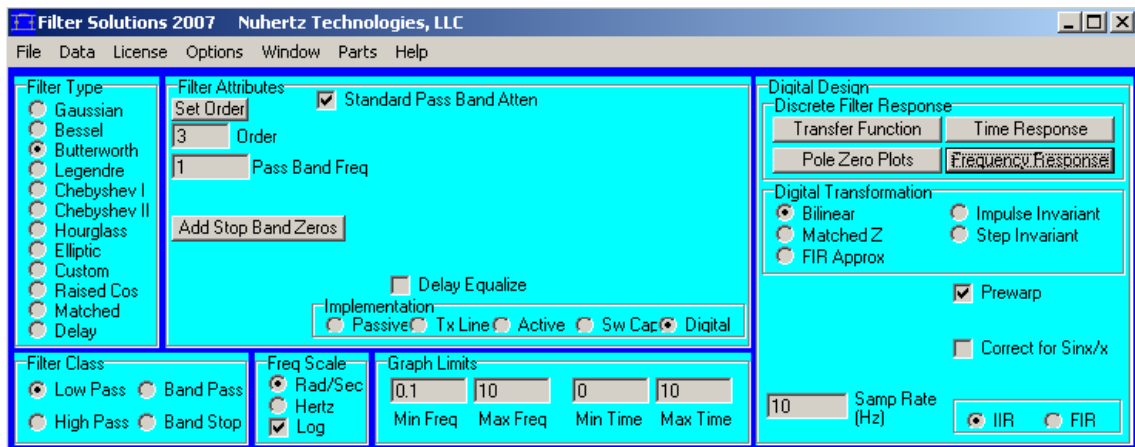
Príklad

Analýzujte príklad overenia navrhnutého IIR filtra (v prostredí Filter Express) pomocou simulácie v Matlabe. Súbor [13] realizuje výpočet amplitúdovej frekvenčnej funkcie IIR filtra realizovaného ako kaskádna sekcia 3 bikvadov. Zároveň je realizovaná simulácia spracovania vstupného signálu tvoreného súčtom dvoch harmonických signálov (jeden z pásma priepustnosti a druhý z pásma tlmenia).

Ďalším programom, ktorý umožňuje okrem napr. návrhu pasívnych a aktívnych analógových filtrov aj číslicové filtre je program Filter Solutions firmy Nuhertz [14], ktorého úvodné grafické rozhranie je zobrazené na Obr.5. Tento program je k dispozícii

⁹ Pre použitím je potrebné získať licenčný kód. Tento je možné získať bezplatne zaslaním emailu na adresu zobrazenú počas inštalácie (február 2005). V súčasnosti táto možnosť nie je dostupná. V laboratóriu však sú k dispozícii funkčné inštalácie s plnou licenciou.

v učebni počas cvičení, pričom zo stránky firmy Nuhertz je možné stiahnuť aj limitované resp. evaluačné verzie. Program patrí medzi špičkové komerčné programy pre návrh filtrov a študentom KEMT je k dispozícii v rámci univerzitného programu firmy Nuhertz.



Obr.5 Grafické rozhranie programu Filter Solutions firmy Nuhertz

1.3 SPEKTRÁLNA ANALÝZA A ALGORITMUS FFT

Diskrétna Fourierova transformácia (DFT) je jednou z najdôležitejších operácií v číslicovom spracovaní signálov [7]. Základným predpokladom využívania DFT je znalosť jej vlastností a efektívnych algoritmov pre jej výpočet. Rýchla Fourierova transformácia (FFT – Fast Fourier Transform) je typickým predstaviteľom rýchleho algoritmu pre výpočet DFT.

Pre komplexnú¹⁰ vstupnú postupnosť

$$x(n) = x_r(n) + jx_i(n) \quad n = 0, 1, \dots, N-1 \quad (1.15)$$

je algoritmus FFT definovaný vzťahom

$$X(k) = X_r(k) + jX_i(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi kn}{N}} \quad k = 0, 1, \dots, N-1 \quad (1.16)$$

t.j. výstupom FFT je vo všeobecnom prípade tiež komplexná postupnosť. Na výpočet FFT v Matlabe sa využíva príkaz **fft**. Na výpočet inverznej FFT (IFFT)

$$x(n) = x_r(n) + jx_i(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi kn}{N}} \quad n = 0, 1, \dots, N-1 \quad (1.17)$$

je možné využiť príkaz **ifft**

Algoritmus FFT sa využíva predovšetkým na výpočet spektra signálu, prípadne na rýchly výpočet konvolúcie vo frekvenčnej oblasti [8][9][10][11]. Keďže spektrum

¹⁰ Samozrejme toto je najvšeobecnejší prípad, vstupná postupnosť môže byť aj rýdzo reálna.

signálu je takmer vždy komplexné, je často potrebné využívať na zobrazenie absolútne hodnoty komplexného čísla (funkcia **abs**).

Príklad

Zobrazte spektrum signálu, ktorý dostaneme ako súčet dvoch sínusoviek s frekvenciami 1500 Hz a 4000 Hz a amplitúdami 3.3, pričom frekvencia vzorkovania je 48000 Hz. Vysvetlite, prečo majú amplitúdy sínusoviek vo frekvenčnej oblasti po výpočte FFT odlišnú veľkosť.

Riešenie

```
n=0:1023; % diskretný čas
x=3.3*sin(2*pi*n*1500/48000)+3.3*sin(2*pi*n*4000/48000);
y=fft(x); % urcenie spektra
plot(abs(y)) % absolutna hodnota komplexneho vektora
```

Príklad

Zobrazte spektrum signálu, na výstupe filtrov navrhnutých v predchádzajúcich príkladoch pre vstupný signál ktorý dostaneme ako súčet dvoch sínusoviek s frekvenciami 1500 Hz a 4000 Hz a amplitúdami 1.0, pričom frekvencia vzorkovania je 48000 Hz. Na výpočet výstupu filtra využite funkciu **filter**.

Riešenie

```
n=0:1023; % diskretný čas
x=1.0*sin(2*pi*n*1500/48000)+1.0*sin(2*pi*n*4000/48000); % generovanie vstupneho signalu
y=filter(b,a,x); % filtracia navrhnutym IIR filtrom
% y=filter(h,1,x); % filtracia navrhnutym FIR filtrom
plot(abs(fft(x))); % zobrazenie spektra vstupneho signalu
plot(y); % zobrazenie vystupneho signalu
plot(abs(fft(y))); % zobrazenie spektra vystupneho signalu
```

LITERATÚRA

- [1] <http://www.analog.com/blackfin> (február 2006).
- [2] <http://www.kemt.fei.tuke.sk/adsp> (február 2006).
- [3] http://www.kemt.fei.tuke.sk/Predmety/KEMT412_SPvT/_materialy/Cvicenia/adsp2181/
- [4] FilterExpress - Digital Filter Synthesis v5.1. Freeware (potrebná registrácia), <http://www.systolix-dsp.com/nf/feintro.htm> (február 2006).
- [5] Filter Design Toolbox. The MathWorks Inc, <http://www.mathworks.com/products/filterdesign/> (február 2006).
- [6] Vích, R.: Transformace Z a některá její použití. SNTL, Praha 1983.
- [7] Čížek, V.: Diskrétní Fourierova transformace a její použití. SNTL, Praha, 1981.

- [8] Ondráček, O.: Signály a systavy. STU Bratislava, 1999.
- [9] Magyar, A.: Číslíkové zpracování signálu II, Signálové procesory a jejich použití. Edičné stredisko TU v Košiciach, 1991.
- [10] Uhlíř, J. – Sovka, P.: Číslíkové zpracování signálu. Vydavatelství ČVUT, Praha 1995.
- [11] Porat, B.: A Course in Digital Signal Processing. John Wiley & Sons, Inc., New York, 1997.
- [12] Burrus, C.S. – McClellan, J.H. – Oppenheim, A.V. – Parks, T.W. – Schaffer, R.W. – Schuessler, H.W.: Computer Based Exercises for Signal Processing using Matlab. Prentice Hall, Englewood Cliffs, New Jersey, 1994.
- [13] testIIR.m – demonštračný príklad, overenie kompletného návrhu filtra IIR pomocou Matlabu, www.kemt.fei.tuke.sk/Predmety/KEMT412_SPVT/_materialy/Cvicenia/testiir.m
- [14] Filter Solutions, <http://www.filter-solutions.com/>