

Úloha č. 11: referát: NE

Návrh a implementace filtru FIR

Úkol cvičení:

Pomocí programu MATLAB navrhnete filtr FIR a implementujete jej na výukové desce DSK Plus firmy Texas Instruments.

Parametry filtru ověříte měřením na osciloskopu a spektrálním analyzátoru.

Parametry filtru

Dolní propust - f_{vst} do 7 kHz, zlomový kmitočet $f_{zlom} = 2$ kHz

Horní propust - f_{vst} do 7 kHz, zlomový kmitočet $f_{zlom} = 3,4$ kHz

Pásmová propust - f_{vst} do 10 kHz, pásmo propustnosti $f_{propust} = 300 - 3400$ Hz

Pásmová zádrž - f_{vst} do 10 kHz, zádržné pásmo $f_{zádrž} = 3$ kHz - 4 kHz

Řád filtru **N volte 80.**

pozn. Pokud byste chtěli zvolit jiný řád filtru, musíte v souboru fir.asm přidat nebo ubrat nuly mezi XN a XNLAST.

Teorie

Amplituda výstupního signálu FIR filtru v čase n je dána konvolutorním součinem:

$$y(n) = h(0) \cdot x(n) + h(1) \cdot x(n-1) + h(2) \cdot x(n-2) + \dots + h(N-1) \cdot x[n - (N-1)],$$

kde $h(n)$ jsou koeficienty filtru, dané impulzní odezvou

$x(n)$ je vstupní vzorek v čase n , $x(n-1) \dots x[n - (N-1)]$ jsou předchozí vzorky

N je řád filtru

Vlastnosti filtru tedy závisí pouze na koeficientech, které se použijí při výpočtu, a program zůstává stejný pro všechny druhy filtrů.

Tyto koeficienty vám vygeneruje program **firkoef**, který spustíte v prostředí MATLAB pod MS Windows.

Funkce programu firkoef:

- provede výpočet koeficientů filtru ze zadaných amplitudových a frekvenčních charakteristik a řádu filtru

Vstupy:

frekvenční vektor ... frekvence normované k Nyquistově frekvenci, tj. polovině vzorkovací f , př. [0 0.2 0.22 1]

amplitudový vektor ... hodnoty v rozmezí 0 a 1 - každé frekvenci přísluší určitá amplituda, př. [1 1 0 0]

řád filtru - N

- vizualizace výsledků - zobrazí amplitudovou a útlumovou frekvenční charakteristiku filtru
- vytvoří soubor **koef.asm**, který uloží na disk do adresáře s FIR filtrem
- přeloží zdrojový program pro výpočet filtru, do kterého je vložen soubor s vypočtenými koeficienty

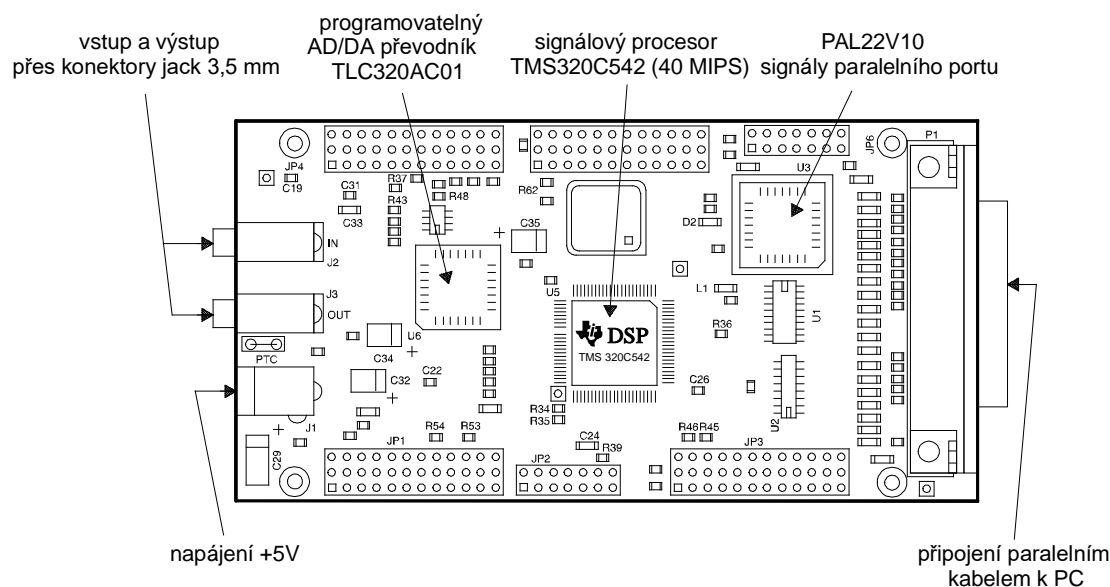
Popis desky DSK Plus

Výuková deska obsahuje vše potřebné pro seznámení se s možnostmi digitálního zpracování signálů. Deska obsahuje moderní digitální signálový procesor TMS320C542, převodník AD/DA a Host Port Interface, který umožňuje desku pomocí paralelního kabelu propojit s počítačem a z něj nahrávat do procesoru vytvořené aplikační programy a v programu Code Explorer běžícím pod MS Windows ladit program a sledovat obsah registrů a paměti procesoru.

Technické parametry desky:

- Procesor TMS320C542 pracující s pevnou řádovou čárkou (16ti bitový)
 - 40 MIPS (instrukční cyklus 25-ns)
 - 10K slov DARAM s dvojím přístupem - možno číst a zapisovat současně (Dual Access RAM)
 - 2K slov zaváděcí ROM
 - 1 sériový port s časovým dělením (TDM)
 - 1 sériový port s automatickým plněním (BSP)
 - 1 host port interface (HPI) pro komunikaci PC <<<>>> DSP
 - 3 režimy se sníženou spotřebou
- Programovatelný AD/DA převodník TLC320AC01 „hlasové kvality“
- Oscilátor 10 MHz
- Možnost připojení emulátoru a dalšího rozšíření (další převodník, paměť...)
- Signály pro paralelní port vyrobeny pomocí PAL22V10
- Vstup a výstup připojen na konektory jack 3,5 mm

Výuková deska DSK plus



Aby filtr správně fungoval, musí se vzorkovací frekvence programovatelného A/D převodníku nastavit tak, aby odpovídala požadovaným parametrům filtru.

Popis převodníku TLC320AC01

Převodník TLC320AC01, který je součástí výukové desky, je převodník “hlasové kvality”. Maximální frekvence vstupního signálu je asi 10 kHz.

Funkci převodníku ovlivňují hodnoty v 8 registrech, z nichž nejdůležitější jsou pro nás první dva, kterými se nastavuje vzorkovací frekvence a určuje rozsah zpracovávaného frekvenčního pásma.

Jsou to **registry A a B**.

Frekvenční rozsah je určen hodnotou uloženou v registru **A**. Tato hodnota určuje taktovací frekvenci filtru se spínanými kapacitami (součást převodníku), pomocí kterého je realizován anti-aliasing filtr. Je to dolní propust, jejíž mezní frekvence se vypočte ze vztahu:

$$f(LP) = FCLK / 40 = 10 \text{ MHz} / (2 * A * 40), \text{ kde } FCLK \text{ je taktovací frekvence filtru se spínanými kapacitami.}$$

$$FCLK = 10 \text{ MHz} / (2 * A)$$

Chceme - li tedy zpracovávat co největší frekvence, snažíme se nastavit co největší FCLK, což znamená A co nejmenší.

Hodnota v A nesmí být menší než 12, což odpovídá $FCLK = 416 \text{ kHz}$ a mezní frekvenci anti-aliasing filtru $10,416 \text{ kHz}$.

Vzorkovací frekvenci ovlivňuje jak hodnota v registru **A**, tak i hodnota v registru **B**.

$$F_s = FCLK / B = 10 \text{ MHz} / (2 * A * B)$$

Hodnota v B nesmí být menší než 10, jinak se nedokončí převod a nastane chyba!

Postup:

1. Požadovaný frekvenční rozsah mi určí hodnotu, kterou mám zapsat do A.
2. Když znám A, mohu pro zvolenou F_s určit hodnotu pro B

Nastavení se týká jen dolních 8 bitů registrů, v hexadecimálním zápisu tedy jen dvou prvních číslic!!!

Možné hodnoty:

A ... **12** - 36 (0Ch - 24h)

B ... **10** - 20 (0Ah - 14h)

Nejjednodušší způsob je hodnoty vyčíst z následující předem vypočtené tabulky:

Hodnoty v tabulce jsou v dekadické soustavě.

registr A	FCLK [kHz]	f(LP) [kHz]	registr B	Fs [kHz]	f(HP) [Hz]
36	138,89	3,47	20	6,94	34,72
			18	7,72	38,58
			15	9,26	46,30
			10	13,89	69,44
32	156,25	3,91	20	7,81	39,06
			18	8,68	43,40
			15	10,42	52,08
			10	15,63	78,13
28	178,57	4,46	20	8,93	44,64
			18	9,92	49,60
			15	11,90	59,52
			10	17,86	89,29
24	208,33	5,21	20	10,42	52,08
			18	11,57	57,87
			15	13,89	69,44
			10	20,83	104,17
20	250,00	6,25	20	12,50	62,50
			18	13,89	69,44
			15	16,67	83,33
			10	25,00	125,00
16	312,50	7,81	20	15,63	78,13
			18	17,36	86,81
			15	20,83	104,17
			10	31,25	156,25
12	416,67	10,42	20	20,83	104,17
			18	23,15	115,74
			15	27,78	138,89
			10	41,67	208,33

Postup při cvičení:

Zkopírujte celý adresář G:\FIR\ do svého domovského adresáře.

Pro daný frekvenční rozsah vstup. signálu vyčtěte z tabulky hodnoty v registrech **A a B**.

Převěďte je do hexadecimální soustavy.

V adresáři s FIR filtrem otevřete soubor **ac01init.asm** a přepište 2 pravé číslice v hexadecimálním zápisu hodnoty v registrech REG1 a REG2 (měním jen dolních 8 bitů registrů A a B).

Př.

REG1 .set 124h

REG2 .set 20fh ; nastaví A na 36 a B na 15

Uložte soubor a spusťte program **firkoef** v MATLABu pod MS Windows. Po zapsání koeficientů na disk zkopírujte z adresáře C:\windows\matlab\bin\ soubor koef.asm do vašeho domovského adresáře s programy FIR a ukončete práci Windows.

Pod DOSem přeložte ve vašem domovském adresáři pomocí DSKPLASM.EXE zdrojového program FIR.ASM (DSKPLASM.EXE FIR.ASM), který vytvoří soubor FIR.OBJ

Spusťte program Code Explorer pod MS Windows.

Pomocí menu FILE >> LOAD PROGRAM nahrajte program FIR.OBJ do signálového procesoru a klepnutím na tlačítko RUN jej spusťte.
Ověřte parametry filtru na osciloskopu a spektrálním analyzátoru.

Informativní výpis hlavního programu FIR.ASM pro implementaci FIR filtru

```
.title    "FIR filter"                ; Tyto ctyri direktivy
        .mmregs                      ; assembleru ovlivnuji
        .width    80                  ; ovlivnuji pouze listing
        .length   55                  ; generovany prekladacem

        .setsect ".text",0x1800,0     ; Tyto direktivy urcuji rozdeleni
        .setsect ".data",0x0200,1     ; pametoveho prostoru RAM na
        .setsect "vectors",0x0180,0   ; programovou a datovou pamet

        .sect "vectors"               ; tabulka vektoru preruseni
        .copy "vectors.asm"           ; zacina na adrese 0x0180

        .sect ".data"                 ; data ulozena na adrese 0x0200 v dat. pameti
XN       .word    0,0,0,0,0,0,0,0,0,0 ; 80 bunek pro ulozeni predchozich
XN2      .word    0,0,0,0,0,0,0,0,0,0 ; vzorku
XN3      .word    0,0,0,0,0,0,0,0,0,0 ;
XN4      .word    0,0,0,0,0,0,0,0,0,0 ;
XN5      .word    0,0,0,0,0,0,0,0,0,0 ;
XN6      .word    0,0,0,0,0,0,0,0,0,0 ;
XN7      .word    0,0,0,0,0,0,0,0,0,0 ;
XNLAST   .word    0                   ;
VYSTUP   .word    0                   ; sem se ulozi vypoctena vyst. hodnota

        .sect ".text"
        goto(start)                  ; jdi na spustitelny kod
        .copy "koef.asm"              ; vloz soubor s koeficienty filtru
        .copy "ac01init.asm"          ; vloz soubor s inicializaci prevodniku

start:   intm = 1                      ; zakaz interrupty
        dcall AC01INIT                ; inicializuj prevodnik
        DP = #0
        nop
        pmst = #01a0h                 ; ukazatel vektoru preruseni na adresu 180h
        sp = #0ffah                   ; ukazatel zasobniku se nachazi v komunikacnim jadre
        imr = #240h                   ; odmaskuj TDM RINT and HPIINT(host port interface)
        intm = 0                      ; umozni vsechny nemaskovane interrupty

WAIT:    goto    WAIT                ; Toto je hlavni program (nekonecna smycka).
        ; Cekam na interrupt vyvolany prichodem hodnoty z prevodniku.
        ; Pri interruptu se provede Interrupt Service Routine (ISR),
        *                                ktera zacina navestim RECEIVE

receive: DP = #XN                     ; Nastavi ukazatel stranky v datove pameti na str. se vst. vzorky
        A = DRR1                      ; Do akumulatoru uloz hodnotu z prevodniku
        @XN = A                       ; Do bunky XN uloz obsah akumulatoru (hodnotu z ADC)
        AR0 = #XNLAST                 ; Do pomocneho registru AR0 natahni adresu posledniho *
        zpozdeného prvku
        repeat(#(XNLAST-XN)), A = #0 ; Opakuj Nkrat a predtim vynuluj akumulator
        macd(*AR0-,h0,A)              ; Vypocti vystup filtru
        ; Vynasobi koeficient a vzorek a pricte k akumulatoru, *
        pote se posune na dalsi koeficient a dalsi vzorek
        *                                Zaroven cele pole vstupnich vzorku posoupne a uvolni *
        tak misto pro další vzorek
        @VYSTUP = hi(A)               ; Uloz vysledek do promenne VYSTUP
        ; Ukladej hornich 16 bitu akumulatoru
        ; do promenne VYSTUP (filtrace)
        A = @VYSTUP                   ; Vystup ==> Akumulator A
        A = #0FFFCh & A               ; 2 nejnizsi bity musi byt 0
        ; Ve skutečnosti je hodnota jen 14 bitu
        ; a 2 LSB urcuji, ze prevodnik je na desce sam
        ; (STAND ALONE).
        DXR1 = A                      ; Vysli hodnotu do D/A prevodniku
        return_enable                 ; Umozni interrupty a vrat se z ISR

transmit:
        return_enable                 ; Umozni interrupty a vrat se z ISR
        .end
```

V souboru **koef.asm** jsou jen koeficienty filtru:

```
h0       .word    10
h1       .word    25
.
.
h79      .word    10
```