

## Realizarea filtrelor digitale

Realizarea digitală a filtrelor prezintă în principal următoarele avantaje:

- eliminarea erorilor datorate variațiilor valorii componentelor cu temperatura și datorate îmbătrânirii componentelor,
- eliminarea erorilor datorate zgomotului indus de sursele de alimentare,
- obținerea unor performanțe ridicate ce nu se pot obține pe cale analogică (deci foarte dificil),
- realizarea unor caracteristici de tăiere foarte abrupte, a unor caracteristici cu amplitudine oarecare (după necesități) și a unor caracteristici de fază liniare.

Dezavantajele filtrelor numerice constau în erorile ce apar datorită cuantizării (atât a semnalului de intrare cât și a elementelor ce caracterizează filtrul numeric-coeficienții filtrului) precum și în apariția ciclurilor limită și a oscilațiilor de depășire (fenomene care apar datorită trunchierii sau rotunjirii rezultatelor parțiale și ca efect al lungimii finite a cuvintelor binare cu care se operează în structura numerică de calcul).

Există două categorii de filtre digitale: filtre cu răspuns finit la impuls (FIR) și filtre cu răspuns infinit la impuls (IIR).

### Realizarea filtrelor FIR

Un filtru cu răspuns finit la impuls (FIR) este un sistem discret liniar și invariant în timp a cărui ieșire este dată de suma ponderată a unui număr finit de intrări întârziate. Filtrele FIR sunt nerecursive (spre deosebire de filtrele IIR); prin urmare nu necesită bucle de reacție (feedback) în calculul lor.

Această proprietate permite o analiză simplă și o implementare cu ajutorul unor procesoare de tipul ADSP 21xx.

Realizarea unui filtru FIR este prezentată în figura 1.

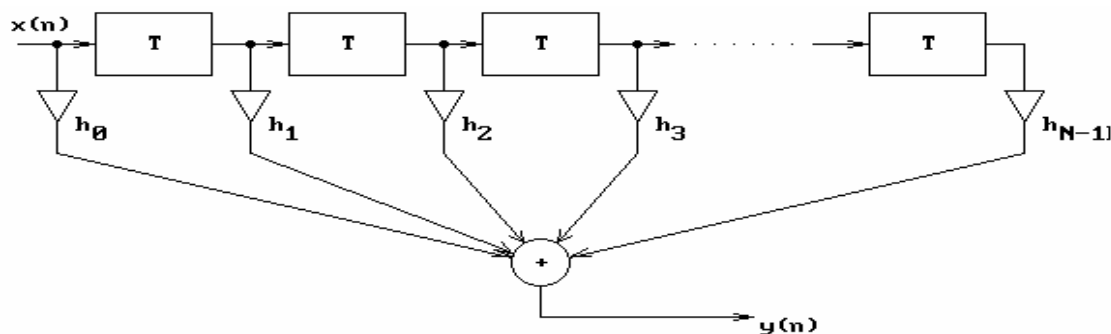


Figura.1 Filtrul nerecursiv FIR

unde  $T$  reprezintă o întârziere cu un pas (tact de eșantionare), iar  $h_k$  sunt coeficienții filtrului FIR calculați cu metoda seriilor Fourier utilizând ferestrele sau metoda Park-McClellan.

Ieșirea filtrului FIR este dată de relația:

$$y(n) = \sum_{k=0, N-1} h_k(n)x(n-k) = \sum_{k=0, N-1} h_k x(n-k) \quad (1)$$

În aceasta relatie s-a pus în evidența faptul ca, coeficientii filtrului  $h_k$  pot sa-și modifice valoarea la fiecare pas  $n$ . În continuare vom presupune ca acești coeficienti sint constanti:  $h_k(n)=h_k$ . În acest caz filtrul nu este un filtru adaptiv.

Pentru implementarea ecuatiei (1) trebuie sa se defineasca doua buffere circulare de lungime  $N$ , unde  $N$  este numarul de intirzieri ale filtrului FIR.

Un buffer circular este folosit pentru a memora intrarea  $X(n)$ , iar celalalt buffer circular memoreaza coeficientii filtrului  $h_k$ .

Situatia este ilustrata în figura 2.

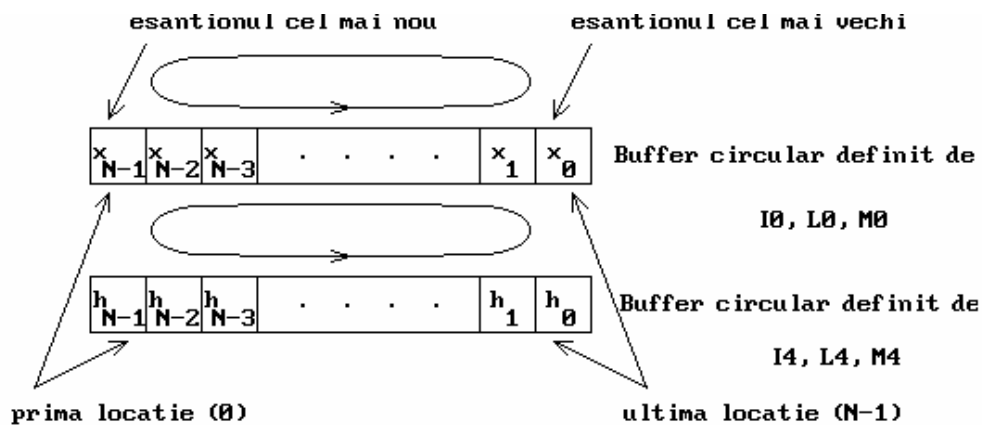


Figura .2. Organizarea bufferelor circulare

Prin inmultirea locatiilor corespondente din cele doua buffere și acumulara rezultatelor se obtine iesirea  $y(n)$ .

Sa consideram un exemplu:  $N=4$  în figura 3.

$$y(0) = \sum_{k=0,3} h_k x(-k) = h_0 x_0 = h_3 x_0$$

$$y(1) = \sum_{k=0,N-1} h_k x(1-k) = h_0 x_1 + h_1 x_0 = h_3 x_1 + h_2 x_0$$

$$y(2) = \sum_{k=0,N-1} h_k x(2-k) = h_0 x_2 + h_1 x_1 + h_2 x_0 = h_3 x_2 + h_2 x_1 + h_1 x_0$$

$$y(3) = \sum_{k=0,N-1} h_k x(3-k) = h_0 x_3 + h_1 x_2 + h_2 x_1 + h_3 x_0 = h_3 x_3 + h_2 x_2 + h_1 x_1 + h_0 x_0$$

$$y(4) = \sum_{k=0,N-1} h_k x(4-k) = h_0 x_4 + h_1 x_3 + h_2 x_2 + h_3 x_1 = h_3 x_4 + h_2 x_3 + h_1 x_2 + h_0 x_1$$

În continuare este prezentata o subrutina în limbaj de asamblare ADSP 2101 care implementează un filtru FIR:

```
.MODULE fir_sub;
{
Parametrii de intrare:
I0 --> Cel mai vechi esantion în linia de întârziere
L0 = Lungimea filtrului FIR (N)
I4 --> Indica începutul tabelii de coeficienti ai FIR
```

L4 = Lungimea filtrului FIR (N)

M1, M5 = 1

CNTR = Lungimea filtrului FIR - 1 (N-1)

Parametrii de iesire:

MR1 = Rezultatul curent al filtrarii

I0 --> Cel mai vechi esantion în linia de întârziere

I4 --> Indica începutul tabelii de coeficienti ai FIR

Toti coeficientii sînt în format 1.15.

}

.ENTRY fir;

fir: MR=0, MX0=DM(I0,M1), MY0=PM(I4,M5);

DO sop UNTIL CE;

sop: MR=MR+MX0\*MY0(SS), MX0=DM(I0,M1), MY0=PM(I4,M5);

MR=MR+MX0\*MY0(RND);

IF MV SAT MR;

RTS;

.ENDMOD;

Aceasta subrutina executa operatiile descrise în figura 3.

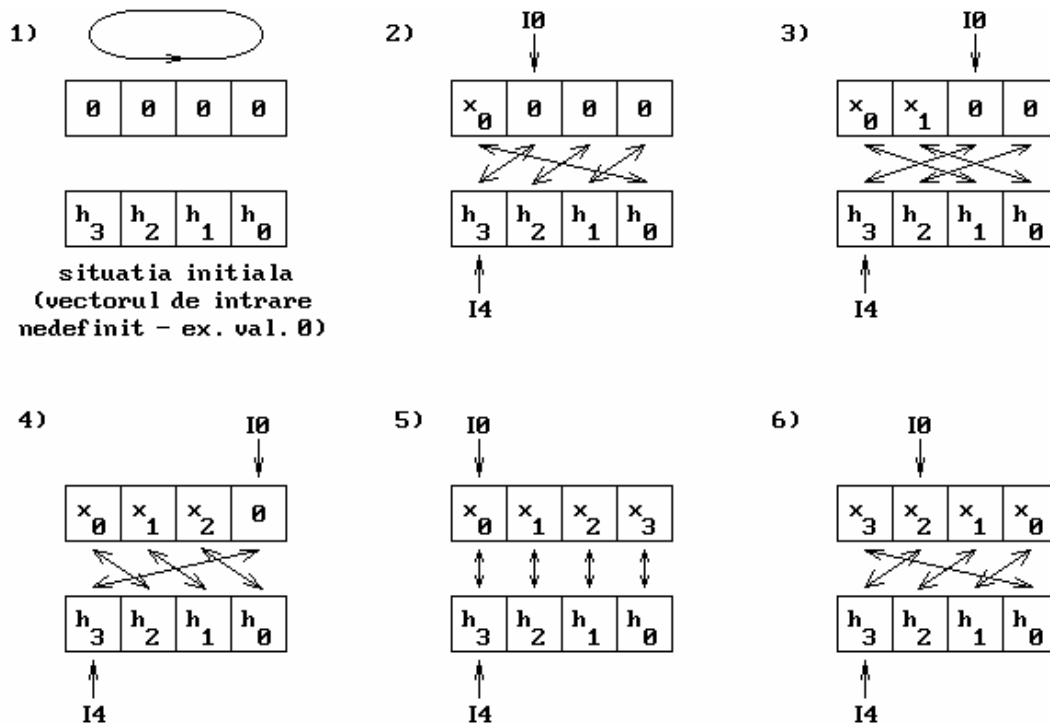


Figura 3 Realizarea prelucrărilor FIR pentru N=4

Săgetile  $\wedge$  indica pointeri I0 și I4, iar săgetile  $\leftrightarrow$  indica locatiile corespondente în cele doua buffere circulare care se vor înmulți și acumula. Se observa ca datorita simetriei coeficientilor putem scrie bufferul de coeficienti în ordine inversa (deci începând cu  $h_0$  și terminând cu  $h_{N-1}$ ).

Initial se sterge registrul MR și se încarca registrele MX0 și MY0 cu primul esantion și respectiv cu primul coeficient al filtrului; aceste valori sint deja stocate în memorie. Se executa apoi de (N-1) ori instructiunea multifunctiede înmulțire cu acumulare și încarcare simultana a 2 date; acest lucru se realizeaza cu ajutorul unei bucle hardware DO-UNTIL CE; initial registrul CNTR se încarca cu N-1. Aceasta bucla calculeaza primele (N-1) nume de produse necesare calculului lui J(n).

Se observa ca dupa fiecare înmulțire cu acumulare se preiau urmatorii 2 operanzi ce trebuie înmulțiti utilizandu-se în acest scop adresarea indirecta cu post-indexare:dupa citirea a 2 operanzi deja este calculata adresa urmatorilor 2; de asemenea bufferele sint circulare ca în figura 3.

Ultima înmulțire se efectueaza separat pentru a permite rotunjirea rezultatului. Daca a aparut depasirea aritmetica rezultatul este saturat.

În acest fel rezultatele intermediare sint acumulate la întreaga rezolutie de 40 biti a registrului MR.

Numarul maxim N de intirzieri ale filtrului FIR, pentru ca implementarea sa fie în timp real este determinat de viteza de lucru a procesorului, perioada de esantionare și de celelalte operatii necesare în afara buclei principale care efectueaza suma de produse.

Este necesar ca timpul total de prelucrare sa fie mai mic sau cel mult egal cu perioada de esantionare.

Pentru subrutina prezentata anterior timpul de prelucrare total este  $T_p = (N-1+5)T_c = (N+4)T_c$  unde  $T_c$  este durata unui ciclu procesor. Luind o valoare tipica  $T_c = 30ns$  și perioada de esantionare  $T_s = 125\mu s$  rezulta  $N = 125\mu s/30ns$  coeficienti; acest numar este suficient de mare pentru implementarea oricarui filtru numeric FIR.