Ministerul Educaţiei, Culturii și Cercetării  
Universitatea Tehnică a Moldovei

Departamentul Informatică și Ingineria Sistemelor

Facultatea Calculatoare, Informatică și Microelectronică

**RAPORT**

Lucrare de laborator Nr.6

la Prelucrarea semnalelor

A efectuat:   
st. gr. TI-206 Mardari Sandu

A verificat:   
asis. univ. Cazac Artiom

Chişinău – 2023

**Mersul lucrării**

Obiective: De studiat proprietăţile de bază ale sistemelor discrete în timp continuu în domeniul de frecvenţă.

**Programul 6\_1**

În acest program se calculează aproximarea de sus. Se foloseşte funcţia sinc din sistemul MATLAB

%Programul P4\_1

% raspunsul impuls filtrului ideal

clf;

fc=0.25;

n=[-6.5:1:6.5];

y=2\*fc\*sinc(2\*fc\*n);

k=n+6.5;

stem(k,y); title('N=13'); axis([0 13 -0.2 0.6]);

xlabel('Timpul n'); ylabel('Amplituda'); grid

Acest program calculeaza aproximarea de sus, unde funcția sinc este utilizată pentru a calcula răspunsul impuls al filtrului. Variabila "fc" reprezintă frecvența de tăiere a filtrului, adică frecvența la care semnalul de intrare este atenuat la jumătate din valoarea sa inițială. Variabila "n" reprezintă eșantioanele timpului. Se calculează răspunsul impuls pentru fiecare eșantion de timp și apoi se afișează grafic folosind funcția "stem" a MATLAB. Aproximarea de sus este o tehnică utilizată pentru a găsi o funcție mai simplă care se apropie cât mai mult de funcția originală.

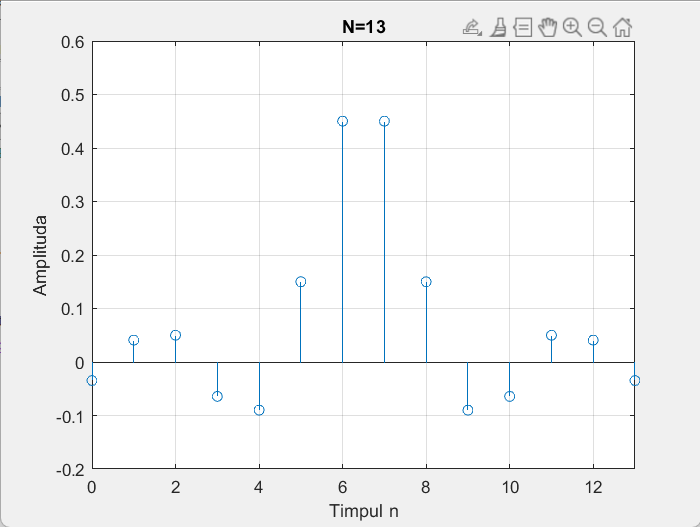


Figura 1 – Graficul pentru aprximarea de sus

**Programul 6\_2**

Programul calculează răspunsul propriu al filtrului de frecvenţă joasă.

% Programul P4\_2

% Rspunsul propriu al filtrului de frecven?? joas?.

clf;

M=2;

num=ones(1,M)/M;

w=0:pi/255:pi;

h=freqz(num,1,w);

g=20\*log10(abs(h));

plot(w/pi,g); grid

axis([0 1 -50 0.5]);

xlabel('\omega /\pi'); ylabel('Adaugare');

title(['M = ',num2str(M)]);

In acest program funcția "freqz" este utilizată pentru a calcula răspunsul în frecvență al filtrului de frecventa. Funcția primește ca parametrii numărătorul "num", numitorul "1", vectorul de frecvențe "w" și returnează valorile răspunsului în frecvență "h". Valoarea absolută a răspunsului în frecvență este calculată apoi utilizând funcția "abs". Pentru a obține o afișare mai intuitivă a răspunsului în frecvență, se aplică o transformare logaritmică la valorile absolute. Transformarea se realizează cu ajutorul funcției "log10", iar valorile astfel obținute sunt scalate cu un factor de 20 pentru a obține unități în decibeli (dB). Aceste valori scalate sunt apoi stocate în variabila "g". Este aplicata formula data astfel încât valoarea rezultată să poată fi mai ușor interpretată și comparată cu alte filtre. Ulterior afisam raspunsul propriu al filtrului pe grafic utilizand functia plot.

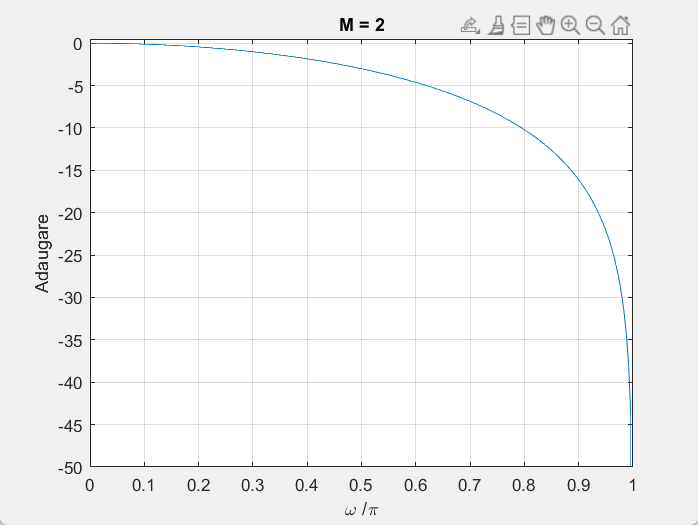


Figura 2 - răspunsul propriu al filtrului de frecvenţă joasă.

**Programul 6\_3**

Cu ajutorul programul dat putem analiza proprietăţile celor patru tipuri de funcţii de transfer care caracterizează filtrele:

Tipul 1: Răspunsul impuls simetric de lungime impară;

Tipul 2: Răspunsul impuls simetric de lungime pară;

Tipul 3: Răspunsul impuls asimetric de lungime impară;

Tiplu 4: Răspunsul impuls asimetric de lungime pară.

% Programul P4\_3

% Punctele zero FIR filtrelor in faza liniara

clf;

b=[1 -8.5 30.5 -63];

num1=[b 81 fliplr(b)];

num2=[b 81 81 fliplr(b)];

num3=[b 0 -fliplr(b)];

num4=[b 81 -81 -fliplr(b)];

n1=0:length(num1)-1;

n2=0:length(num2)-1;

subplot(2,2,1); stem(n1,num1);

xlabel('Timpul n'); ylabel('Amplituda'); grid

title('Tipul 1 FIR filtrului');

subplot(2,2,2); stem(n2,num2);

xlabel('Timpul n'); ylabel('Amplituda'); grid

title('Tip 2 FIR filtrului');

subplot(2,2,3); stem(n1,num3);

xlabel('Timpul n'); ylabel('Amplituda'); grid

title(' Tip 3 FIR filtrului ');

subplot(2,2,4); stem(n2,num4);

xlabel('Timpul n'); ylabel('Amplituda'); grid

title('Tip 2 FIR filtrului');

pause

subplot(2,2,1); zplane(num1,1);

title(' Tip 1 FIR filtrului ');

subplot(2,2,2); zplane(num2,1);

title(' Tip 2 FIR filtrului ');

subplot(2,2,3); zplane(num3,1);

title('Tip 3 FIR filtrului');

subplot(2,2,4); zplane(num4,1);

title('Tip 4 FIR filtrului');

disp('Zeroul FIR filtrului de Tip 1');

disp(roots(num1));

disp(' Zeroul FIR filtrului de Tip 2');

disp(roots(num2));

disp(' Zeroul FIR filtrului de Tip 3');

disp(roots(num3));

disp(' Zeroul FIR filtrului de Tip 4');

disp(roots(num4));

Programul dat afiseaza grafic cele 4 tipuri de FIR ale filtrului. Initial cream vectorul b cu coeficientii filtrului in faza liniara. Apoi pentru fiecare tip de filtru cream un vector cu coeficienti care este format din concatenarea vectorului b si inversa acestuia. Pentru ca se cere simetric impar am adaugat si coeficientul 81 la mijloc, iar pentru simetric par am adaugat 2 valori de 81. In asa mod am procedat si pentru tipurile asimtrice. Apoi acesti vectori sunt afisati grafic utilizand functia stem.

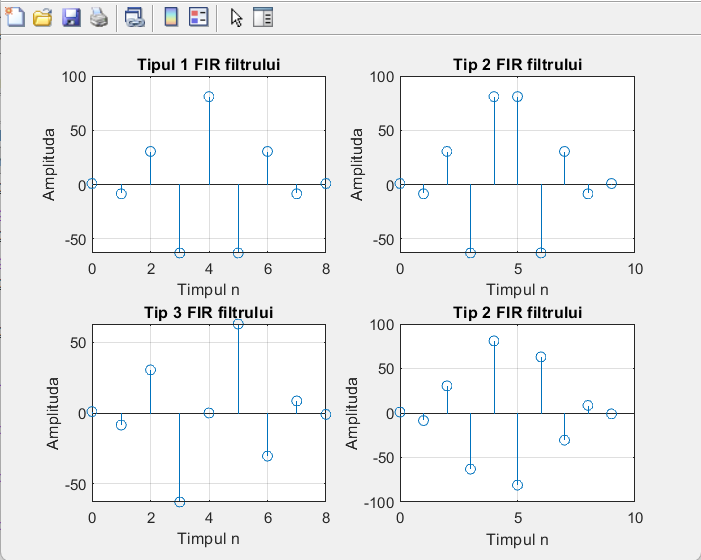


Figura 3 – Cele 4 tipuri de filtre

În continuare, programul utilizează comanda zplane pentru a afișa punctele zero ale filtrului în planul complex. Aceste puncte zero reprezintă locațiile în planul complex în care funcția de transfer a filtrului este zero. Prin urmare, ele joacă un rol important în determinarea caracteristicilor de filtrare ale unui filtru.

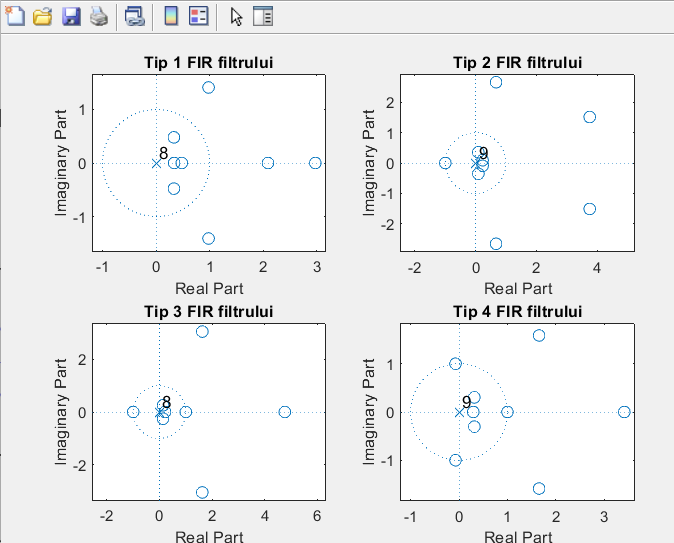


Figura 4 – Cele 4 tipuri de filtre prezentate in plan complex

Zeroul FIR filtrului de Tip 1

2.9744 + 0.0000i

2.0888 + 0.0000i

0.9790 + 1.4110i

0.9790 - 1.4110i

0.3319 + 0.4784i

0.3319 - 0.4784i

0.4787 + 0.0000i

0.3362 + 0.0000i

Zeroul FIR filtrului de Tip 2

3.7585 + 1.5147i

3.7585 - 1.5147i

0.6733 + 2.6623i

0.6733 - 2.6623i

-1.0000 + 0.0000i

0.0893 + 0.3530i

0.0893 - 0.3530i

0.2289 + 0.0922i

0.2289 - 0.0922i

Zeroul FIR filtrului de Tip 3

4.7627 + 0.0000i

1.6279 + 3.0565i

1.6279 - 3.0565i

-1.0000 + 0.0000i

1.0000 + 0.0000i

0.1357 + 0.2549i

0.1357 - 0.2549i

0.2100 + 0.0000i

Zeroul FIR filtrului de Tip 4

3.4139 + 0.0000i

1.6541 + 1.5813i

1.6541 - 1.5813i

-0.0733 + 0.9973i

-0.0733 - 0.9973i

1.0000 + 0.0000i

0.3159 + 0.3020i

0.3159 - 0.3020i

0.2929 + 0.0000i

**Programul 6\_4**

Acest program prezintă cercetarea stabilităţii a filtrului numeric IIR.

Stabilitatea filtrului − reprezintă o calitate foarte importantă a filtrului. Filtrul numeric IIR este stabil dacă polii funcţiei de transfer se află în interiorul cercului unitate.

% Programul P4\_4

% Test de verificare a stabilitatii

clf;

den=input('Introduceti coeficientii de numitor:');

ki=poly2rc(den);

disp('Parametrii testului de stabilitate: ');

disp(ki);

if max(abs(ki)) < 1

fprintf('Filtrul este stabil.\n');

else

fprintf('Filtrul nu este stabil.\n');

end

zplane(roots(den));

Acest program testează stabilitatea unui filtru dat de coeficienții de numitor ai funcției de transfer și afișează graficul polilor și al zerourilor folosind funcția zplane. În primul rând trebuie să introducem coeficienții de numitor ai funcției de transfer (den). Api utilizam funcția poly2rc pentru a calcula rădăcinile polinomului. Rădăcinile polinomului reprezintă polii funcției de transfer și sunt stocate în vectorul ki. Apoi, programul afișează valorile parametrilor testului de stabilitate și verifică dacă valoarea maximă absolută a parametrilor este mai mică decât 1. Dacă această condiție este îndeplinită, programul afișează un mesaj care confirmă că filtrul este stabil, altfel, afișează un mesaj care confirmă că filtrul nu este stabil. În cele din urmă, funcția zplane este utilizată pentru a afișa graficul polilor și al zerourilor funcției de transfer bazată pe rădăcinile polinomului de numitor (den).

>> Program\_4

Introduceti coeficientii de numitor:[10 2]

Parametrii testului de stabilitate:

0.2000

Filtrul este stabil.

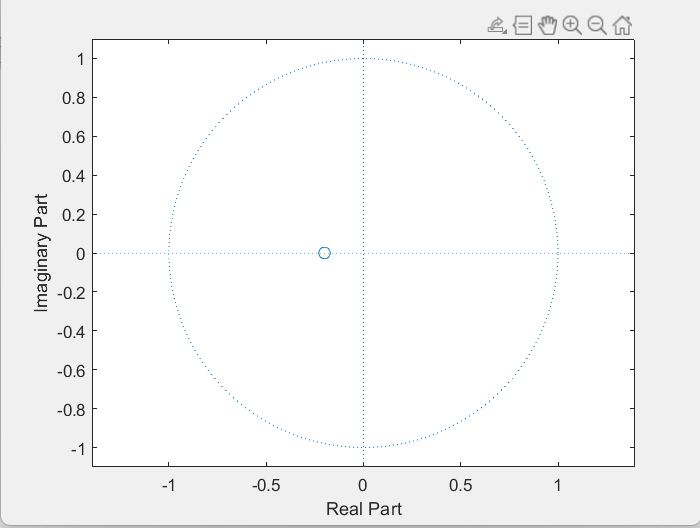


Figura 5 – Filtru stabil

>> Program\_4

Introduceti coeficientii de numitor:[1 10]

Parametrii testului de stabilitate:

10

Filtrul nu este stabil.

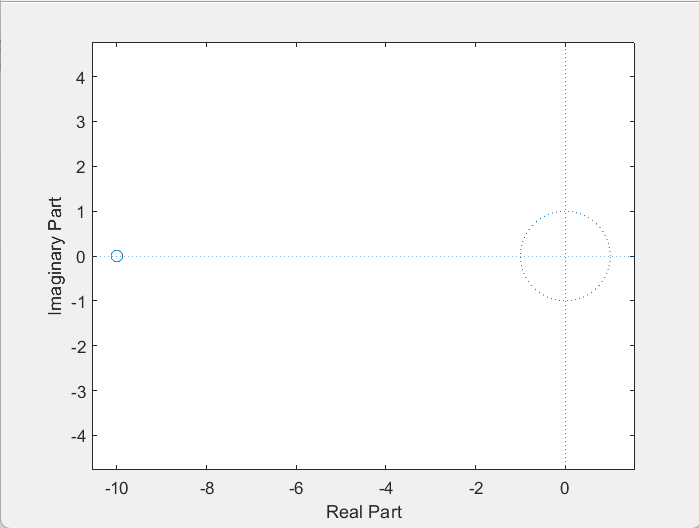


Figura 6 – Filtru instabil

Concluzii:  
 In cadrul acestei lucrari de laborator am analizat patru programe si graficele generate de acestea. Cele patru programe prezentate au ilustrat câteva aspecte ale filtrării digitale, pornind de la definirea unui filtru de frecvență joasă și calculul răspunsului său propriu, trecând prin prezentarea tipurilor de filtre FIR și terminând cu testul de verificare a stabilității unui filtru. Programele prezentate au arătat că în filtrarea digitală, procesul de definire și proiectare a unui filtru este important pentru obținerea unor rezultate bune. În plus, este necesar să se acorde atenție aspectelor precum stabilitatea filtrului, care poate fi verificată prin testarea coeficienților săi de numitor. În general, filtrarea digitală poate fi o tehnică puternică pentru analizarea și prelucrarea semnalelor, și poate fi utilizată într-o varietate de domenii, cum ar fi telecomunicațiile, prelucrarea de imagini și de semnale biomedicale Programele prezentate ne-au demonstrat că în filtrarea digitală, procesul de definire și proiectare a unui filtru este crucial pentru obținerea rezultatelor dorite. Configurarea corectă a parametrilor și alegerea adecvată a tipului de filtru pot avea un impact semnificativ asupra performanței filtrului și asupra calității semnalului filtrat rezultat. De asemenea, am învățat că stabilitatea filtrului este un aspect important ce trebuie luat în considerare. Verificarea coeficienților de numitor ne permite să evaluăm stabilitatea filtrului și să ne asigurăm că acesta nu introduce oscilații nedorite sau instabilitate în semnalul filtrat. În concluzie, această lucrare de laborator ne-a oferit o perspectivă asupra diferitelor aspecte ale filtrării digitale. Am înțeles că proiectarea corectă a unui filtru și verificarea stabilității acestuia sunt cruciale pentru obținerea unor rezultate precise și fiabile în analiza și prelucrarea semnalelor. Filtrarea digitală este o tehnică puternică și versatilă, cu aplicații în diverse domenii, cum ar fi telecomunicațiile, prelucrarea imaginilor și semnalelor biomedicale. Cunoștințele acumulate în cadrul acestei lucrări de laborator ne vor fi utile în explorarea și dezvoltarea ulterioară a acestui domeniu fascinant..