5Ministerul Educației și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Facultatea Calculatoare, Informatică și Microelectronică

Departamentul Ingineria Software și Automatică

**RAPORT**

**Lucrarea de laborator nr.5**

**la Prelucrarea Semnalelor**

*Tema: Sisteme discrete în timp continuu sau discret.*

**Grupa academică:**  TI-211

**A efectuat:**  Bucșan Daniel

**A verificat:**  Potlog Mihail

Chișinău 2024

**Exercițiul 1**

Prezentarea unui filtru simplu cu M-vârfuri, numărul cărora se introduce de către utillizator.

Codul sursă

% Programul P2\_1

% Simularea filtrului FIR de o lungime arbitrara

% generarea semnalului de intrare

clf; n=0:100;

s1=cos(2\*pi\*0.05\*n); % semnal de o frecventa joasa

s2=cos(2\*pi\*0.47\*n); % semnal de o frecventa inalta

x=s1+s2;

% Realiyarea filtrului FIR

M=input('Lungimea dorita a filtrului = ');

num=ones(1,M);

y=filter(num,1,x)/M;

% Afisarea semnalelor de intare si de isire

subplot(2,2,1);

plot(n,s1);

axis([0, 100, -2, 2]);

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

title('semnal #1');

subplot(2,2,2);

plot(n,s2);

axis([0, 100, -2, 2]);

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

title('semnal #2');

subplot(2,2,3);

plot(n,x);

axis([0, 100, -2, 2]);

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

title('Semnalul de intrare');

subplot(2,2,4);

plot(n,y);

axis([0, 100, -2, 2]);

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

title('Semnalul de iesire');

axis;

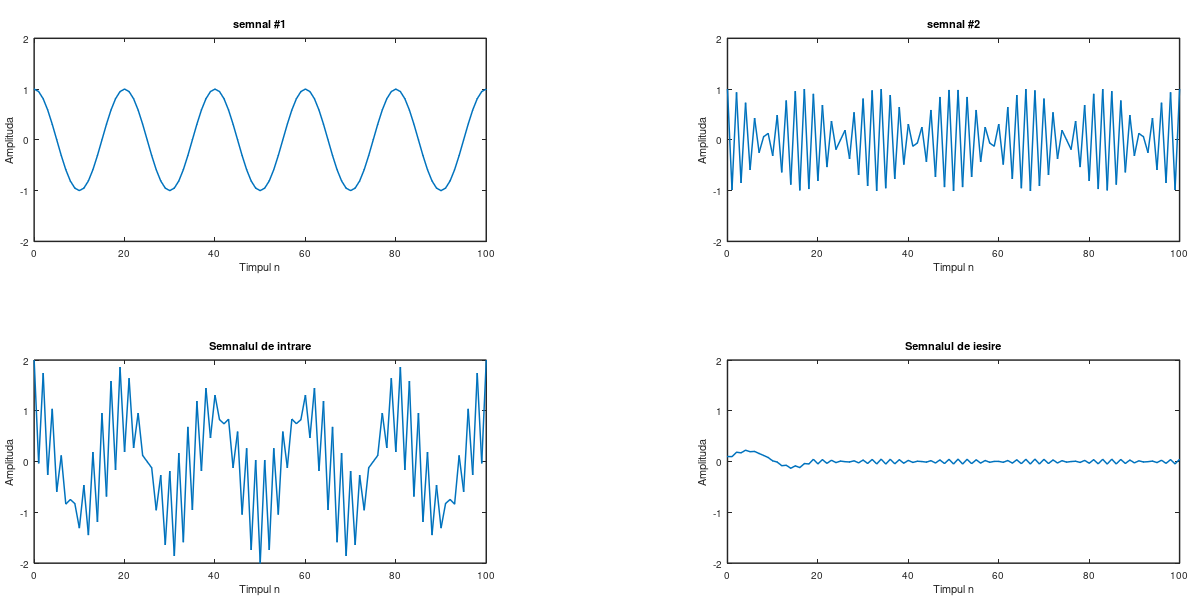


Figura 1 – Filtru simplu

**Exercițiu 2**

Programul de mai jos poate fi folosit pentru generarea semnalului de intrare x[n], compus dintr-o secentă sinusoidală.

% Programul P2\_2

% generarea semnalului sinusoidal de intrare

clf;

n=0:200;

x=cos(2\*pi\*0.05\*n);

% Calcularea semnalului de iesire

x1 = [x 0 0]; % x1[n] = x[n+1]

x2 = [0 x 0]; % x2[n] = x[n]

x3 = [0 0 x]; % x3[n] = x[n-1]

y=x2.\*x2 + x1.\*x3;

y=y(2:202);

% Vizualizarea semnalului de intre si a celui de iesire

subplot(2,1,1);

plot(n,x);

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

title('semnalul de intrare ');

subplot(2,1,2);

plot(n,y);

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

title('Semnalul de iesire');

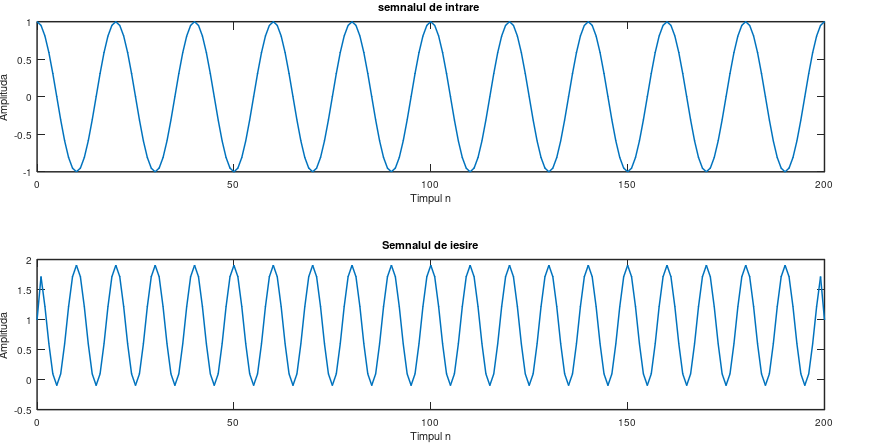


Figura 2 – Semnal sinusoidal

**Exercițiu 3**

În programul acesta este realizat sistemul reprezentat prin următoarea expresie:

y[n]-0.4y[n - 1] + 0.75y [n – 2] = 2.2403x[ n ] + 2.4908x[n – 1] + 2.2403x[n – 2].

Se generează trei secvențe de intrare x1[n], x2[n] si x[n] = a\*x1[n] + b\*x2[n]. Se calculează trei secvențe de ieșire y1[n], y2[n] si y[n], cu reprezentarea loc grafică.

% Programul P2\_3

% Generarea secventelor de iesire

clf;

n=0:40;

a=2; b=-3;

x1=cos(2\*pi\*0.1\*n);

x2=cos(2\*pi\*0.4\*n);

x=a\*x1+b\*x2;

num=[2.2403 2.4908 2.2403];

den=[1 -0.4 0.75];

ic=[0 0]; % initializarea

y1=filter(num,den,x1,ic); % Calcularea semnalului de iesire y1[n]

y2=filter(num,den,x2,ic); % Calcularea semnalului de iesire y2[n]

y=filter(num,den,x,ic); % Calcularea semnalului de iesire y[n]

yt=a\*y1+b\*y2;

d=y-yt; % calcularea abaterii d[n]

%Vizualizarea semnalelor de iesire si a semnalului de abatere

subplot(3,1,1);

stem(n,y);

ylabel('Amplituda ');

title('Semnalul de iesire, conform semnalului de intrare incarcat : a \cdot x\_{1}[n] + b \cdot x\_{2}[n]');

subplot(3,1,2);

stem(n,yt);

ylabel('Amplituda');

title('semnalul de iesire incarcat : a \cdot y\_{1}[n] + b \cdot y\_{2}[n]');

subplot(3,1,3);

stem(n,d);

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

title('semnalul diferential ');

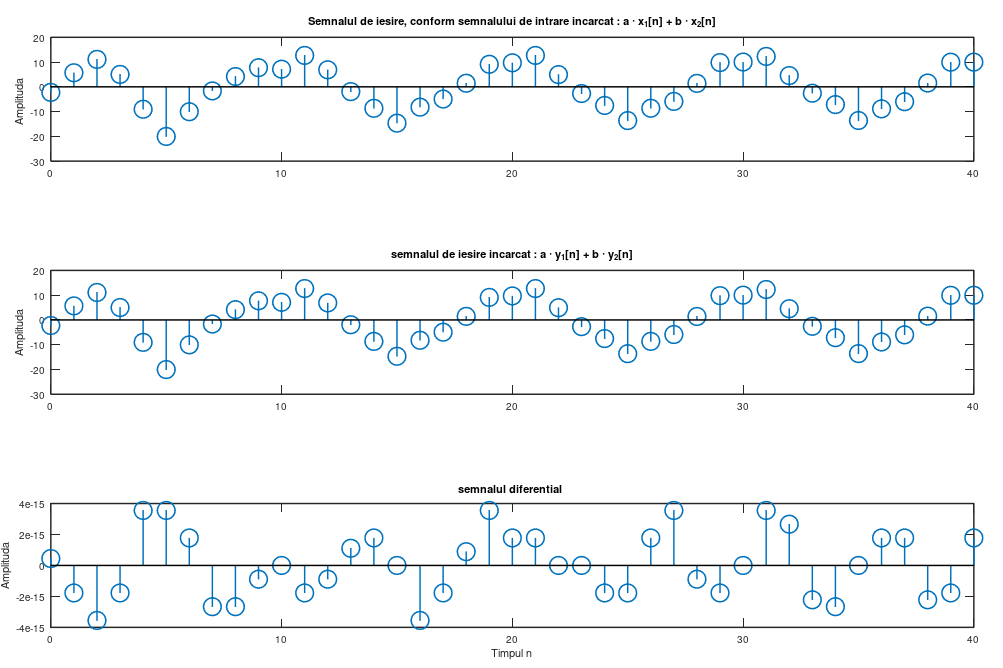


Figura 3 – Semnal de ieșire

**Exercițiu 4**

Pentru această expresie se crează sistemul ce o simulează: y[n]-0.4y[n-1]+0.75y[n-2]=2.2403x[n]+2.4908x[n-1]+2.2403x[n-2]. Două secvenţe de intrare diferite x[n] şi x[n-D]. Se calculează şi se vizualizează corespunzător doua secvenzede ieşire şi abaterea y1[n] – y2[n+D].

% Programul P2\_4

% Generarea secventelor de intrare

clf; n=0:40; D=10; a=3.0; b=-2;

x=a\*cos(2\*pi\*0.1\*n)+b\*cos(2\*pi\*0.4\*n);

xd=[zeros(1,D) x];

num=[2.2403 2.4908 2.2403]; den=[1 -0.4 0.75];

ic=[0 0]; % setarea conditiilor initiale

%Calcularea semnalului de iesire y[n]

y=filter(num,den,x,ic);

% Calcularea semnalului de iesire yd[n]

yd=filter(num,den,xd,ic);

% Calcularea semnalului deabatere d[n]

d=y-yd(1+D:41+D);

%Afisarea graficelor semnalelor de iesire

subplot(3,1,1);

stem(n,y);

ylabel('Amplituda');

title('Semnalul de iesire y[n]'); grid;

subplot(3,1,2);

stem(n,yd(1:41));

ylabel('Amplituda');

title(['iesirea,la intirzierea semnalului la intrare x[n',num2str(D),']']); grid;

subplot(3,1,3);

stem(n,d);

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

title('semnalul de abatere'); grid;

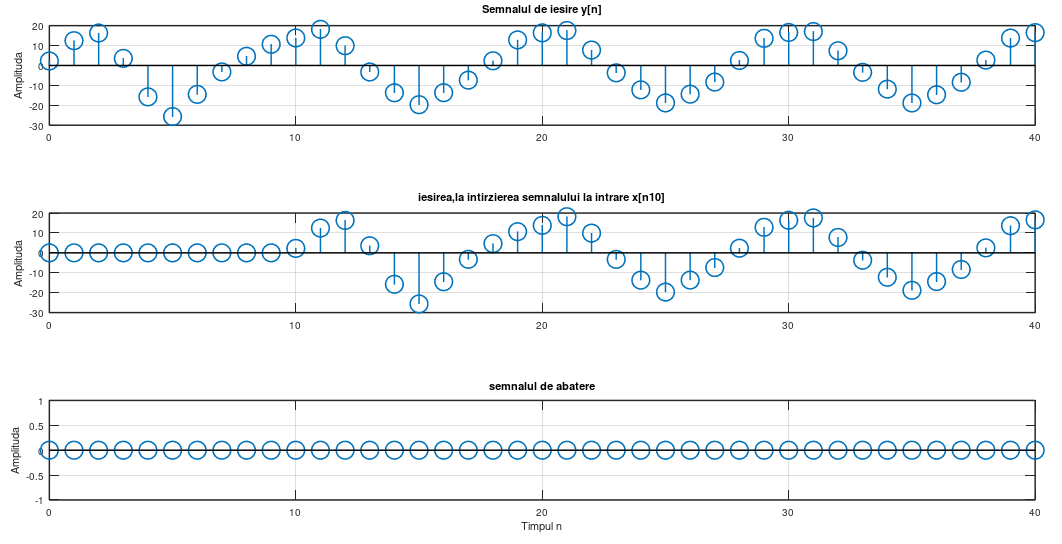


Figura 4 – Semnal de ieșire și de abatere

**Exercițiu 5**

Aici se calculează şi se afişează la ecran impulsul răspunsului sistemului, corespunzător expresiei:y[n]-0.4y[n-1]+0.75y[n-2]=2.2403x[n]+2.4908x[n-1]+2.2403x[n-2].

% Programul P2\_5

% calcularea impulsului de raspuns y

clf;

N=40;

num=[2.2403 2.4908 2.2403];

den=[1 -0.4 0.75];

y=impz(num,den,N);

% Reprezentareagrafică a impulsului de raspuns

stem(y);

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

title('Impulsul de raspuns'); grid;

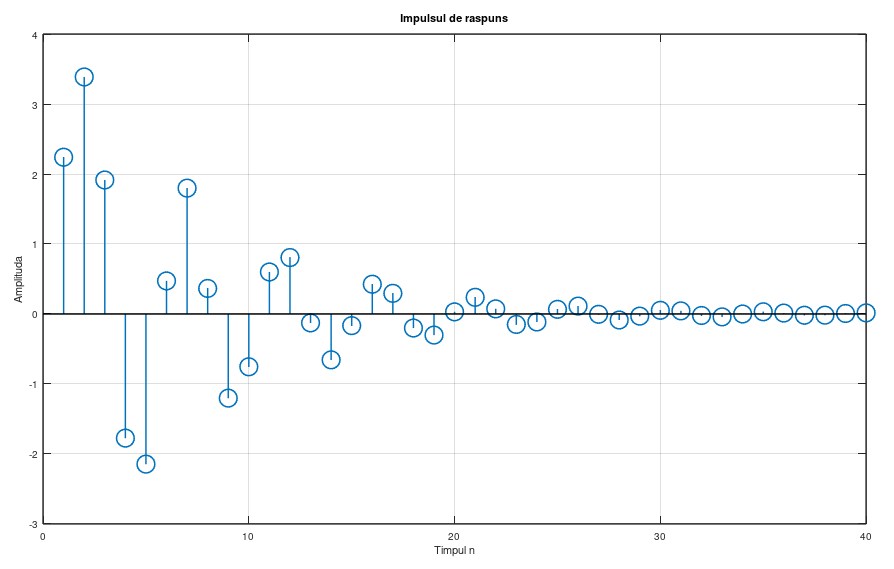


Figura 5 – Impuls de răspuns

**Exercițiul 6**

Programul acesta realizează sistemul de 4 conditii: y[n]+1.6y[n-1]+2.28y[n-2]+1.325y[n3]+0.68y[n-4] = 0.06x[n]-0.19x[n-1]+0.27x[n-2]-0.26x[n-3]+0.12x[n-4] şi Sisteme cascade : Etapa 1: y1[n]+0.9y1[n-1]+0.8y1[n-2]=0.3x[n]-0.3x[n-1]+0.4x[n-2]; Etapa 2: y2[n]+0.7y2[n-1]+0.85y2[n-2]=0.2y1[n]-0.5y1[n-1]+0.3y1[n-2];

% Programul P2\_6

% Realizarea cascadelor

clf;

x=[1 zeros(1,40)]; % Generarea semnalului de intrare

n=0:40;

% Coeficientii sistemului de gradul 4

den=[1 1.6 2.28 1.325 0.68];

num=[0.06 -0.19 0.27 -0.26 0.12];

% calcularea semnalului de ieasire a sistemului de gradul 4

y=filter(num,den,x);

% Coeficientii sistemelor de gradul 2

num1=[0.3 -0.2 0.4]; den1=[1 0.9 0.8];

num2=[0.2 -0.5 0.3]; den2=[1 0.7 0.85];

% Semnalul de iesire y1[n] a primei etape a cascadei

y1=filter(num1,den1,x);

% Semnalul de iesire y2[n] etapei a doua a cascadei

y2=filter(num2,den2,y1);

% abaterea dintre y[n] si y2[n]

d=y-y2;

% Graficele semnalelor de iesire si a abaterii

subplot(3,1,1);

stem(n,y);

ylabel('Amplituda');

title('afisarea realizarii de gradul 4'); grid;

subplot(3,1,2);

stem(n,y2);

ylabel('Amplituda');

title('Afisarea realizarii cascada'); grid;

subplot(3,1,3);

stem(n,d);

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

title('semnalul abaterii'); grid;

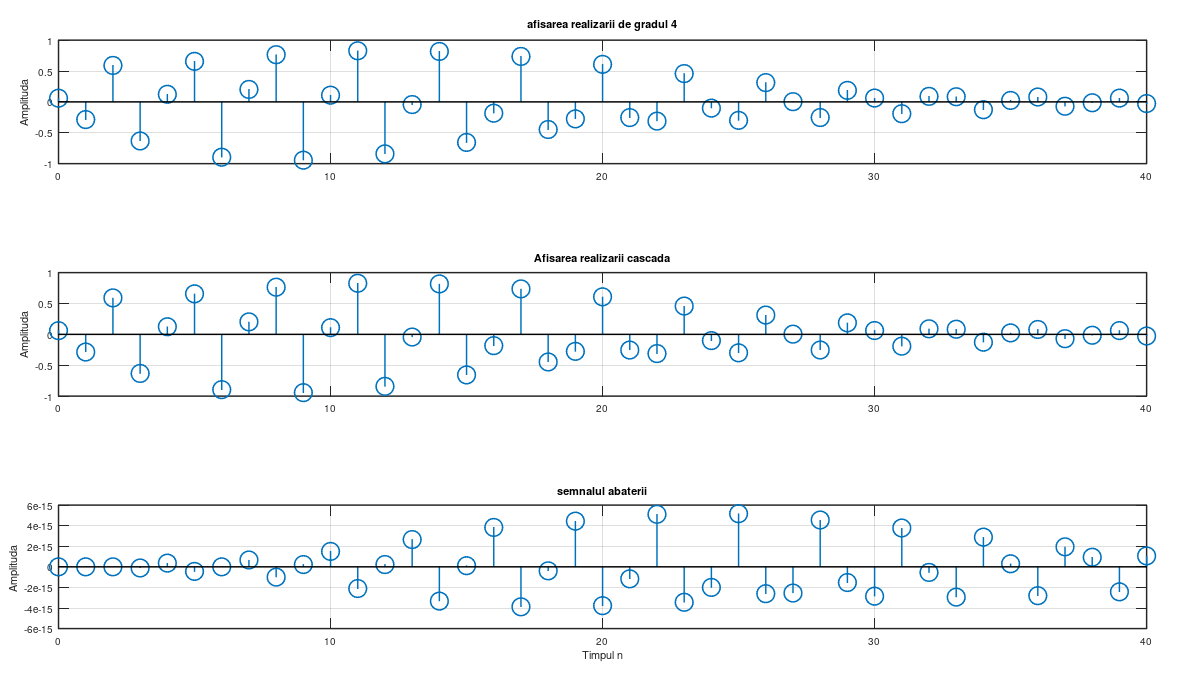


Figura 6 – Sistemul de 4 condiții și sisteme cascade

**Exercițiul 7**

Operaţiunea de convoluţie se realizează în MATLAB cu comanda conv. Pentru aceasta se folosesc două secvenţe finite şi de aceeaşi lungime.

% Programul P2\_7

clf;

h=[3 2 1 -2 1 0 -4 0 3]; % raspuns de impuls

x=[1 -2 3 -4 3 2 1]; % secventa deintrare

y=conv(h,x);

n=0:14;

subplot(2,1,1);

stem(n,y);

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

title('Semnalul de iesire,obtinut in urma convolutiei'); grid;

x1=[x zeros(1,8)];

y1=filter(h,1,x1);

subplot(2,1,2);

stem(n,y1);

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

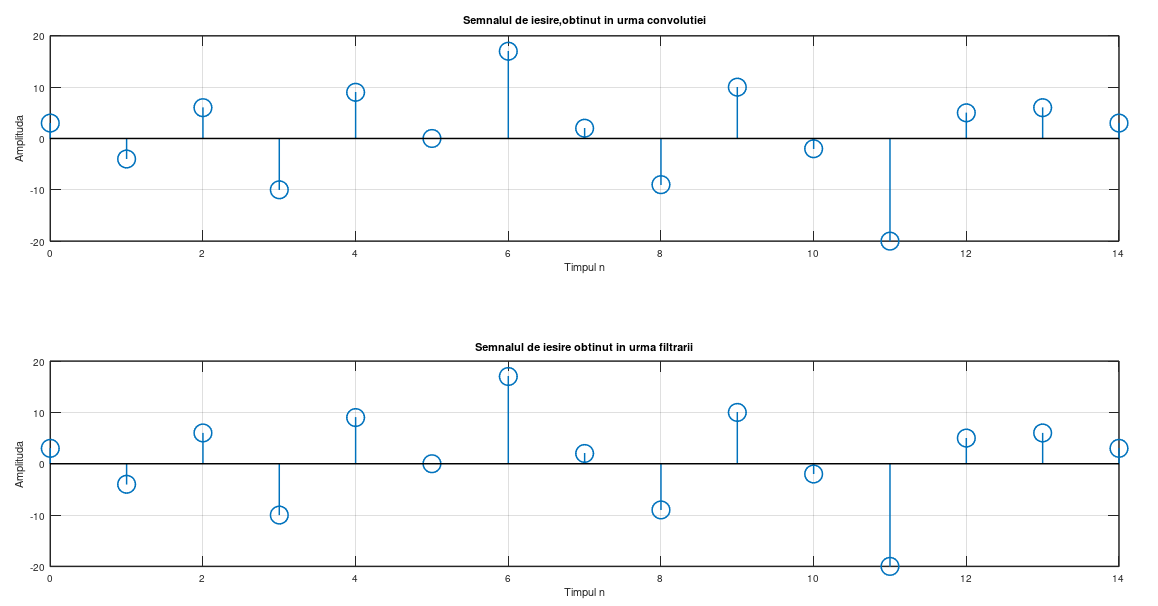
title('Semnalul de iesire obtinut in urma filtrarii'); grid; 

Figura 7 – Realizarea convoluției

**Exercițiul 8**

Programul calculează suma valorilor absolute ale raspunsului inpuls.

% Programul P2\_8

% Testarea stabilitatii bazat pe suma valorilor absolute ale elementelor raspuns impuls

clf;

num=[1 -0.8]; den=[1 1.5 0.9];

N=200;

h=impz(num,den,N+1);

parsum=0;

for k=1:N+1;

parsum=parsum+abs(h(k));

if abs(h(k))<10^(-6), break, end

end

% Afisarea raspuns impuls

n=0:N;

stem(n,h);

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

disp('Valoarea='); disp(abs(h(k))); %Afisarea valorilor

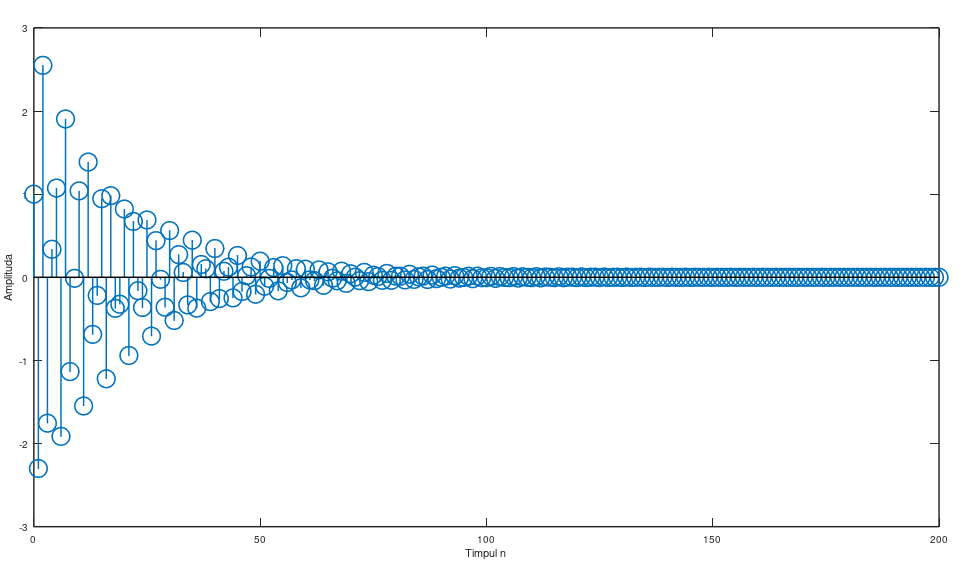


Figura 8 – Suma valorilor absolute ale răspunsului inpuls

**Exercițiul 9**

Aici se prezintă două sisteme: y[n] = 0.5x[n]+0.27x[n-1]+0.77x[n-2] şi y[n] = 0.45x[n] +0.5x[n-1]+0.45x[n-2]+0.53y[n-1]-0.46y[n-2].

% Programul P2\_9

% Generarea semnalului de intrare

clf;

n=0:299;

x1=cos(2\*pi\*10\*n/256);

x2=cos(2\*pi\*100\*n/256);

x=x1+x2;

% Calcularea secventelor de iesire

num1=[0.5 0.27 0.77];

y1=filter(num1,1,x); % Iesirea sistemului #1

den2=[1 -0.53 0.46];

num2=[0.45 0.5 0.45];

y2=filter(num2,den2,x); % Iesirea sistemului #2

% Graficele secventelor de iesire

subplot(2,1,1);

plot(n,y1); axis([0 300 -2 2]);

ylabel('Amplituda');

title('semnalul de iesire #1'); grid;

subplot(2,1,2);

plot(n,y2); axis([0 300 -2 2]);

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

title('semnalul de iesire #2'); grid;

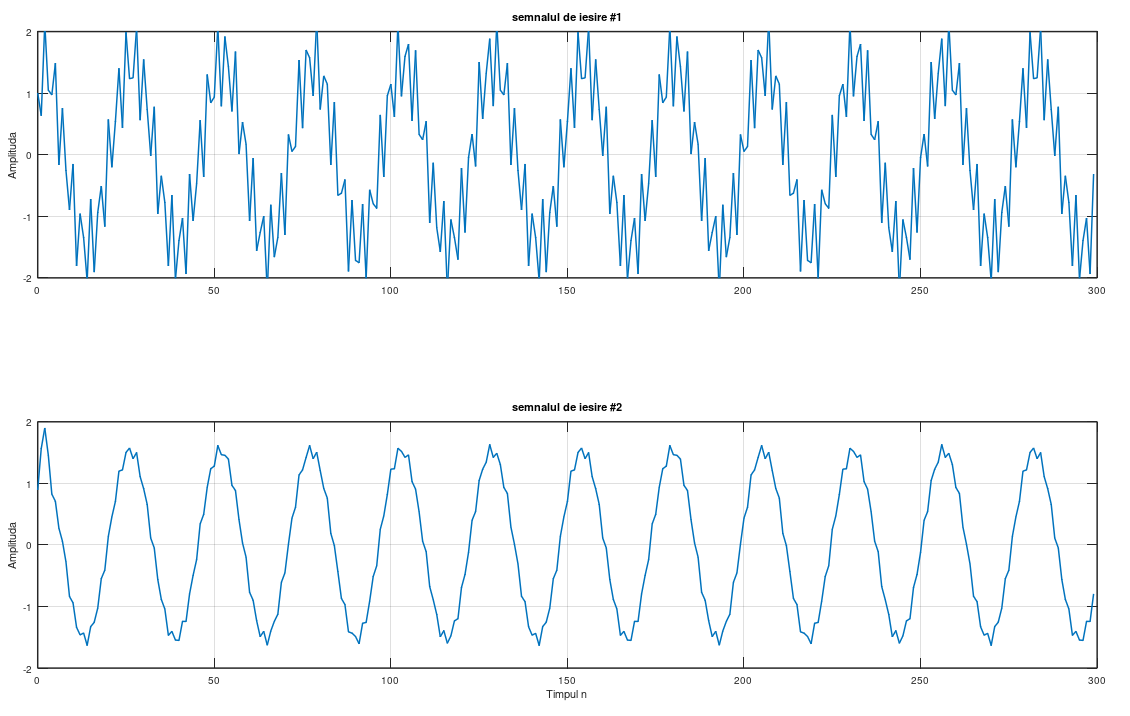


Figura 9 – Reprezentarea a două sisteme

**Concluzii**

Laboratorul a acoperit diverse aspecte ale prelucrării semnalelor. În primul exercițiu, am implementat un filtru FIR simplu cu o lungime dată de utilizator, iar în exercițiile următoare am explorat diverse aspecte precum generarea și prelucrarea semnalelor, convoluția, testarea stabilității și analiza sistemelor. Fiecare exercițiu a contribuit la înțelegerea conceptelor de bază și aplicarea practică a acestora în prelucrarea semnalelor. În concluzie, laboratorul a oferit o experiență cuprinzătoare în domeniul prelucrării semnalelor, evidențiind importanța în simularea și analiza sistemelor de semnale.