Ministerul Educației, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnica a Moldovei

Facultatea Calculatoare, Informatică și Microelectronică

Departamentul Ingineria Software și Automatică

**Raport**

Lucrarea de laborator Nr. 4

la Prelucrarea semnalelor

Tema: Eșantionarea și cuantizarea semnalelor.

Interpolarea semnalelor eșantionate.

|  |  |
| --- | --- |
| A efectuat | St. gr. TI-206  Pleșu Cătălin |
|  |  |
| A verificat | asist. univ.  Cazac Artiom |

Chișinău 2023

**Obiective**: înțelegerea conceptelor de eșantionare, cuantizare, de zgomot de cuantizare și interpolare a semnalelor eșantionate.

Ex 1.1: Reprezentați grafic 1024 de eșantioane ale unui semnal alcătuit din 2 sinusoide (una cu frecvența de 50 Hz, defazajul 0 și amplitudinea 0.5 V, iar cealaltă cu frecvența de 230 Hz, defazajul π/3 și amplitudinea 0.2 V), folosind o frecvență de eșantionare de 8 kHz.

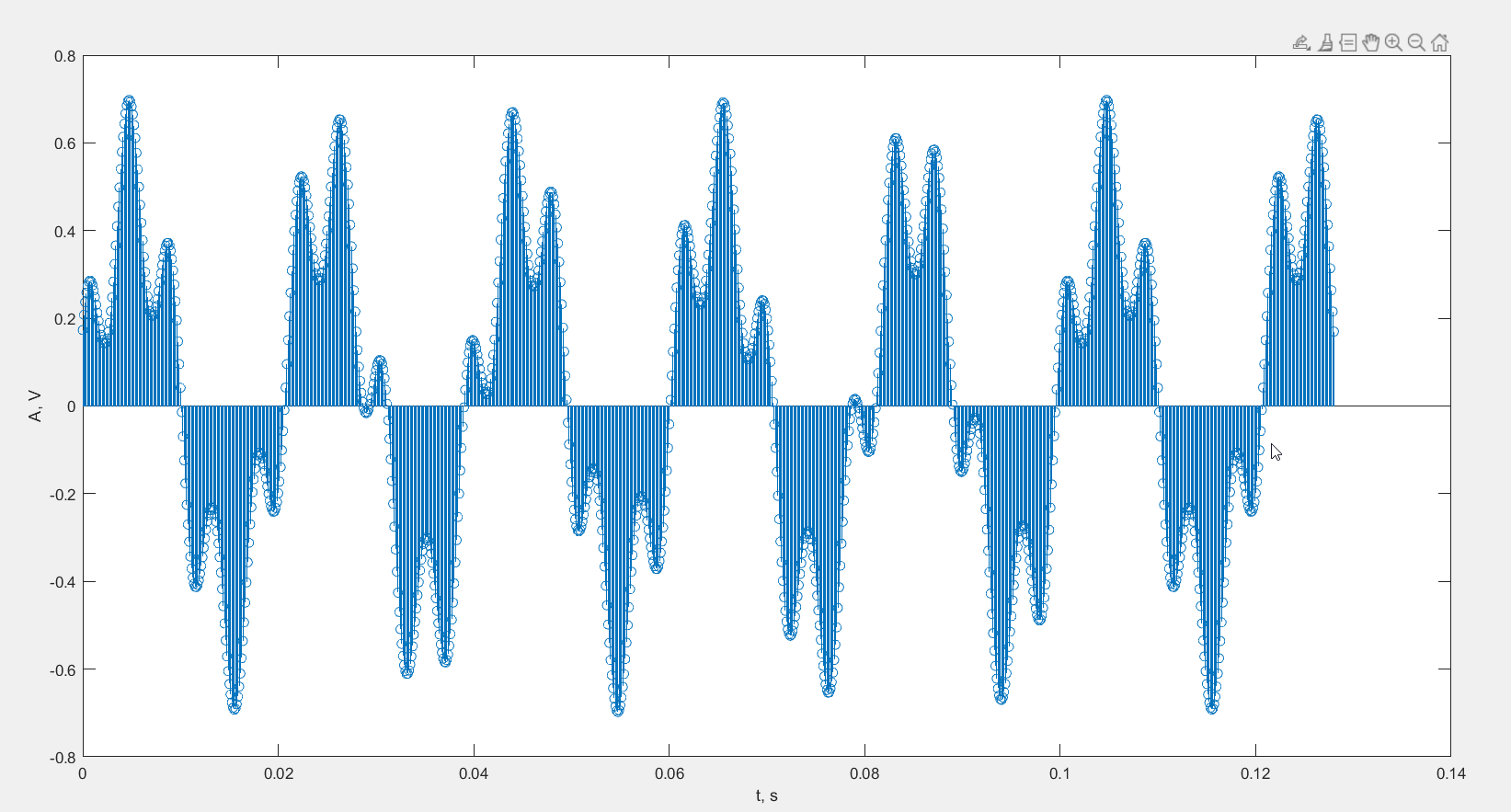
fs=8000;

t = 0 : 1/fs : 1024/fs;

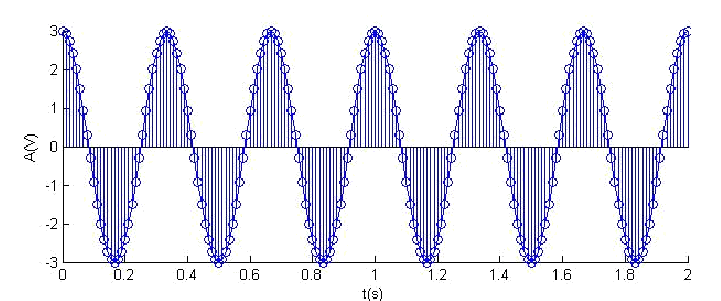
s1 = 0.5\*sin(2\*pi\*50\*t);

s2 = 0.2\*sin(2\*pi\*230\*t+pi/3);

stem(t,s1 + s2);



Ex 1.2: Să se genereze în MATLAB semnalul din figura de mai jos. Indiciu: pornind de la figură, trebuie să se identifice toți parametrii sinusoidei (amplitudine, frecvență, frecvență de eșantionare, durată, fază inițială). Frecvența fs a fost aleasă astfel, încât să fie 30 de eșantioane într-o perioadă.

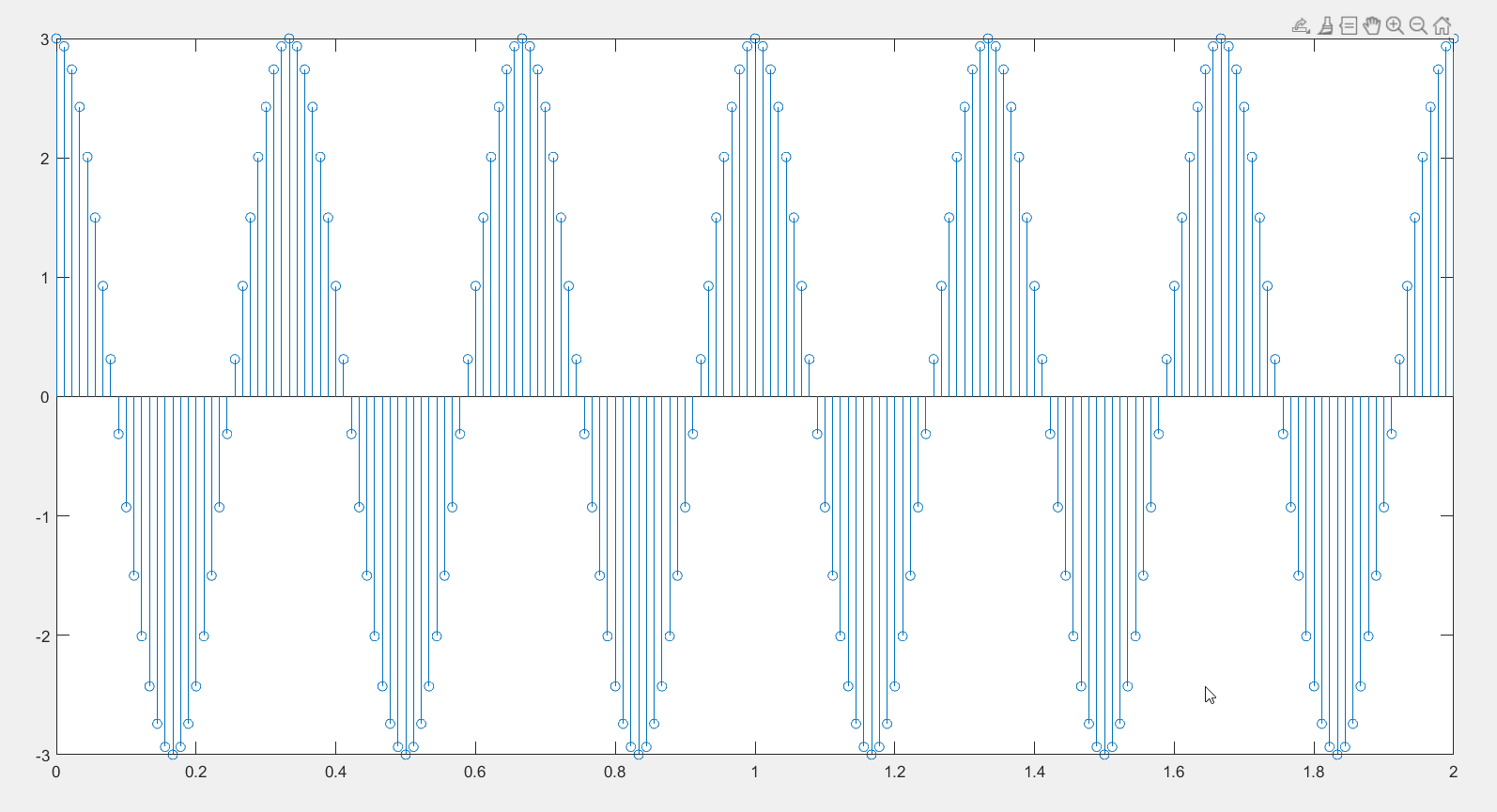


fs=90;

t = 0 : 1/fs : 2;

x = 3\*cos(2\*pi\*3\*t);

stem(t,x);



Ex 1.3: Fie semnalul:

x(t) = 10 sin (200πt +π/2 ) + 20 sin(100πt) - 40sin (300πt -π/4 ).

Cerințe:

care este frecvența minimă de eșantionare astfel încât să se respecte teorema eșantionării;

fs ≥ 2 \* fm ≥ 2 \* 150 ≥ 400 Hz

alegând o frecvență de eșantionare de 10 ori mai mare decât cea determinată la punctul anterior, să se eșantioneze semnalul x(t) și să se reprezinte graphic

care este frecvența de repetiție a semnalului x(t)

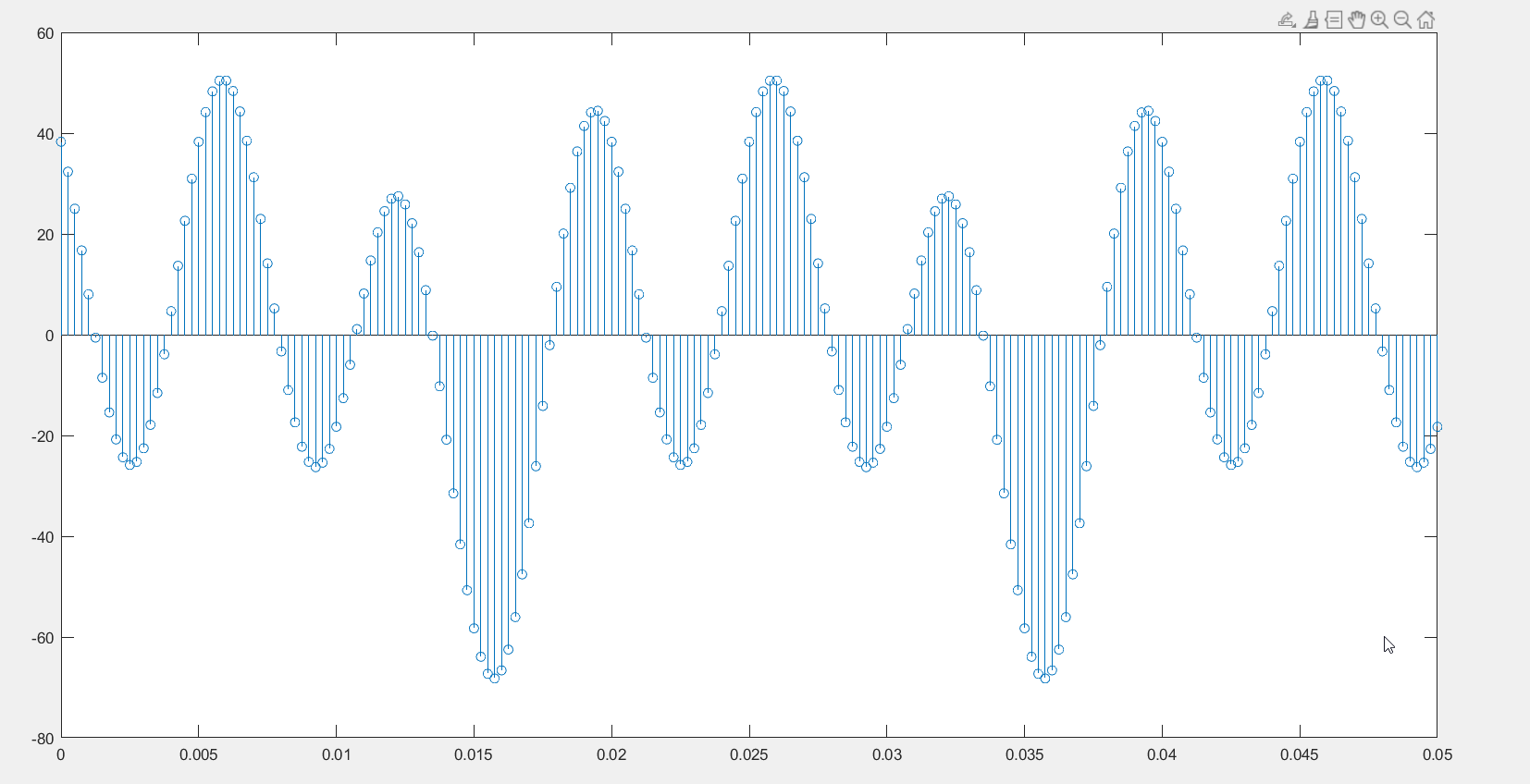
50hz

fs=4000;

t = 0 : 1/fs : 200/fs;

x = 10\*sin(pi\*200\*t+pi/2)+20\*sin(pi\*100\*t)-40\*sin(pi\*300\*t-pi/4);

stem(t,x);



Ex 1.4: Fie semnalul sinusoidal 𝑥(𝑡) = 3sin(2𝜋 ∙ 5𝑡). Să se eșantioneze acest semnal cu fs1 = 4Hz și cu fs2 = 50Hz și să se reprezinte grafic. În care dintre cele două cazuri poate fi reconstituit semnalul x(t) din eșantioanele sale?

fs=4;

t = 0 : 1/fs : 2;

x = 3\*sin(2\*pi\*5\*t);

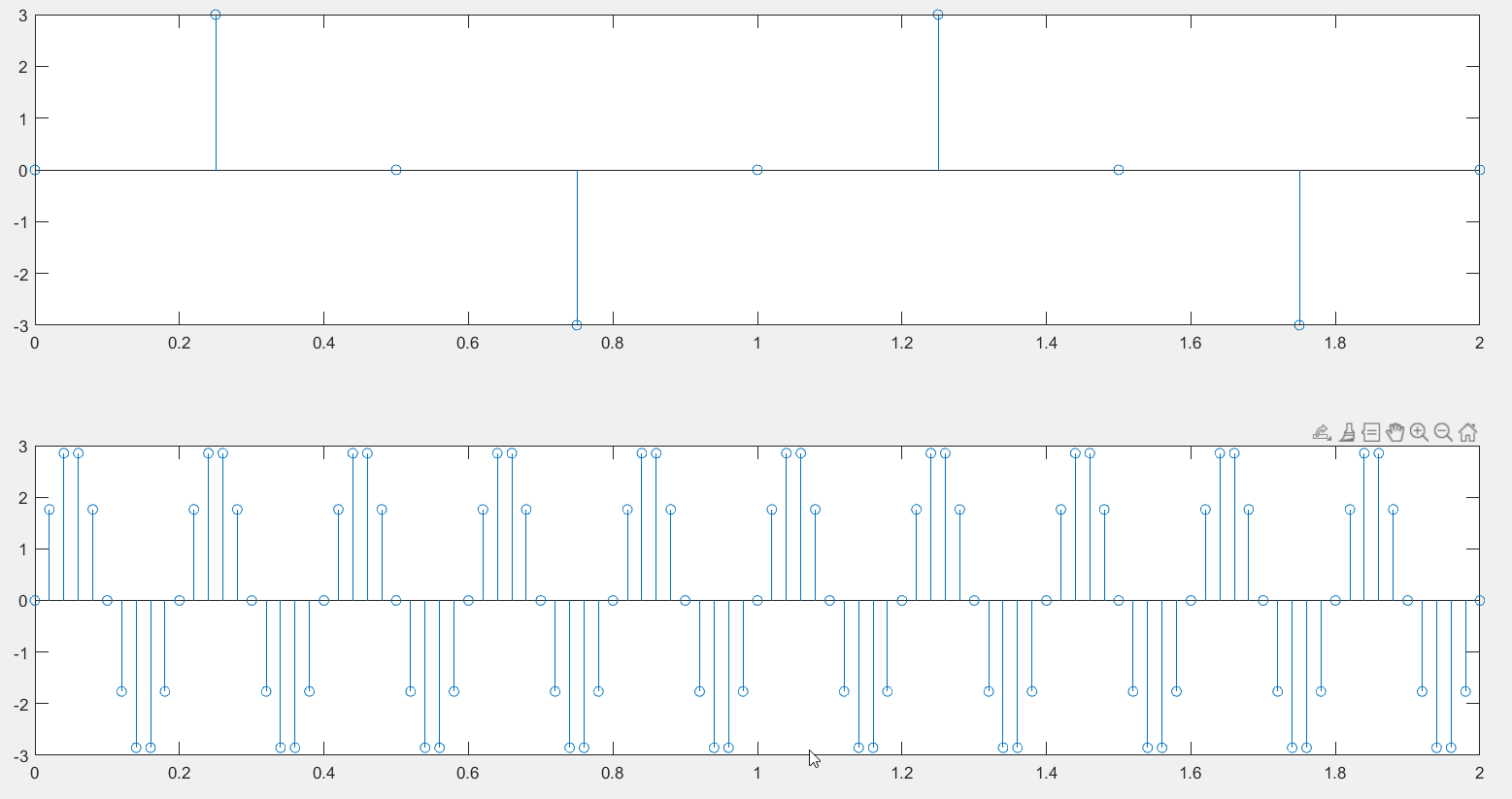
subplot(2,1,1); stem(t,x);

fs=50;

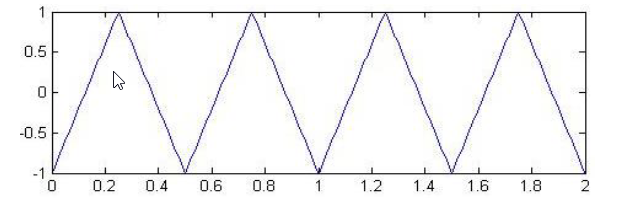
t = 0 : 1/fs : 2;

x = 3\*sin(2\*pi\*5\*t);

subplot(2,1,2); stem(t,x);



Ex 1.5: Să se genereze și să se prezinte în MATLAB, în două ferestre separate, semnalul din figura următoare și semnalul discretizat.



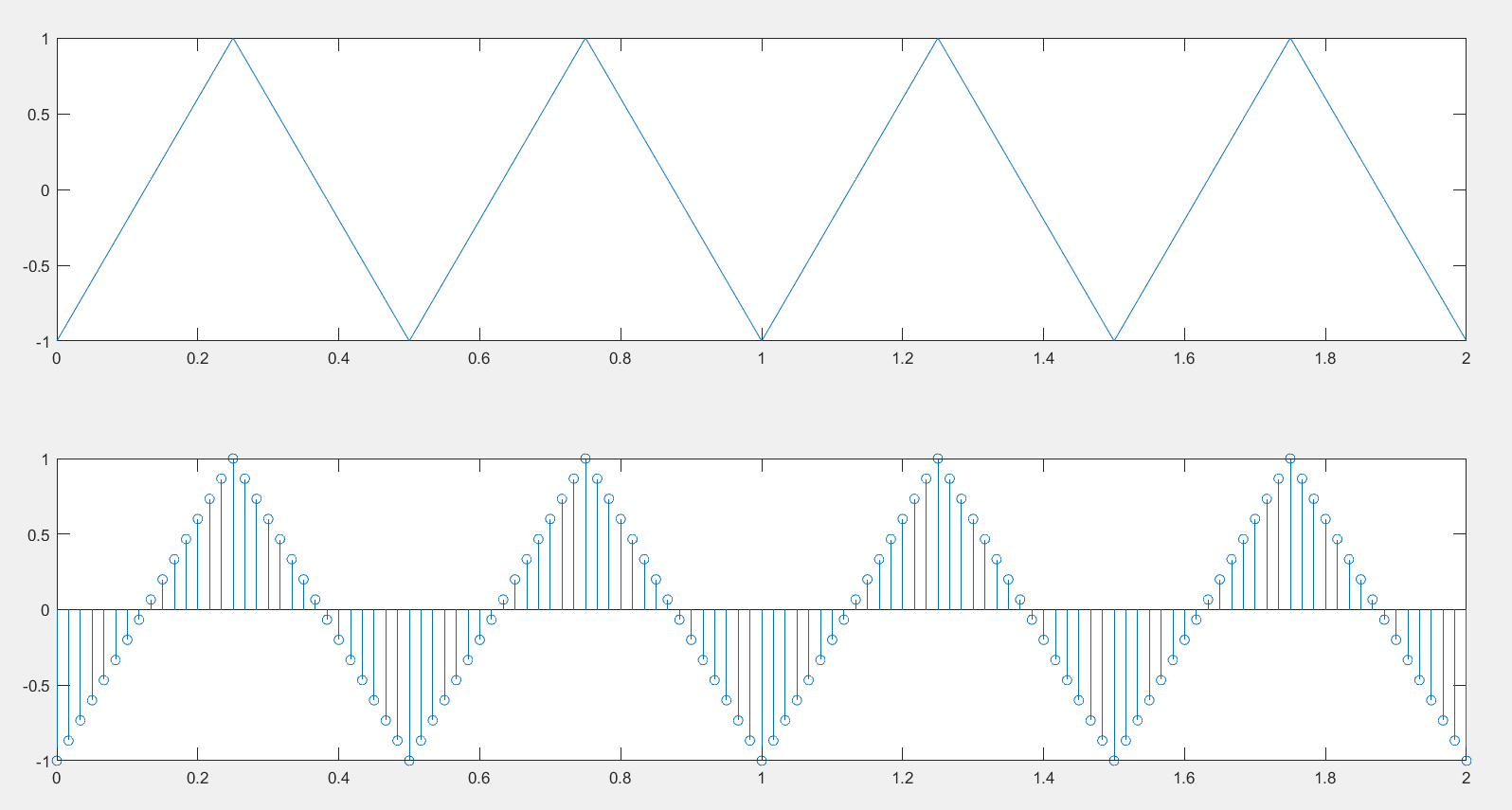
Indiciu: pornind de la figură, trebuie să se identifice toți parametrii semnalului triunghiular (amplitudine, frecvență, frecvență de eșantionare, durată). Frecvența fs se alege astfel încât să fie 30 de eșantioane într-o perioadă.

t=0:1/60:2;

x=sawtooth(4\*pi\*t,0.5);

subplot(2,1,1), plot(t,x);

subplot(2,1,2), stem(t,x);



Ex 2

1: Un semnal cu durata de 2 minute este eșantionat cu frecvența de eșantionare de 4kHz. Câte eșantioane vor rezulta? Dacă fiecare eșantion este stocat pe 2 octeți, ce memorie vor ocupa toate eșantionale generate?

nr=4000\*120=480000

m=2\*480000=960000 Bytes

2: Să se cuantizeze semnalele 𝑥[𝑛] = 17/𝑛 și 𝑦[𝑛] = − 17/𝑛 , 𝑛 = 1 … 40, folosind metodele floor(trunchiere) și round(rotungire). Se cunosc nivelurile de cuantizare 1 {0, ±1, ±2, … , ±17}. Să se reprezinte grafic semnalele originale 𝑥[𝑛] și 𝑦[𝑛], precum și semnalele obținute în urma cuantizării. Să se calculeze și să se reprezinte grafic zgomotul de cuantizare.

n = 1:0.01:40;

x=17./n;

figure(1)

x1 = round(x);

subplot(2,1,1), plot(n,x,n,x1);

xlabel('n'); ylabel('x[n]'); title('Semnalul cuantizat (metoda round)');

z=x-x1;

subplot(2,1,2), plot(n,z);

xlabel('n'); ylabel('x[n]-xq[n]'); title('Zgomot (metoda round)');

figure(2)

x1 = floor(x);

subplot(2,1,1), plot(n,x,n,x1);

xlabel('n'); ylabel('x[n]'); title('Semnalul cuantizat (metoda floor)');

z=x-x1;

subplot(2,1,2), plot(n,z);

xlabel('n'); ylabel('x[n]-xq[n]'); title('Zgomot (metoda floor)');

y=-17./n;

figure(3)

y1 = round(y);

subplot(2,1,1), plot(n,y,n,y1);

xlabel('n'); ylabel('x[n]'); title('Semnalul cuantizat (metoda round)');

z=y-y1;

subplot(2,1,2), plot(n,z);

xlabel('n'); ylabel('x[n]-xq[n]'); title('Zgomot (metoda round)');

figure(4)

y1 = floor(y);

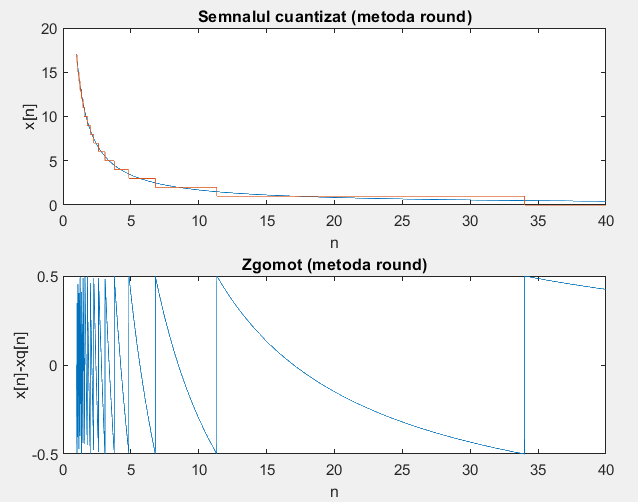
subplot(2,1,1), plot(n,y,n,y1);

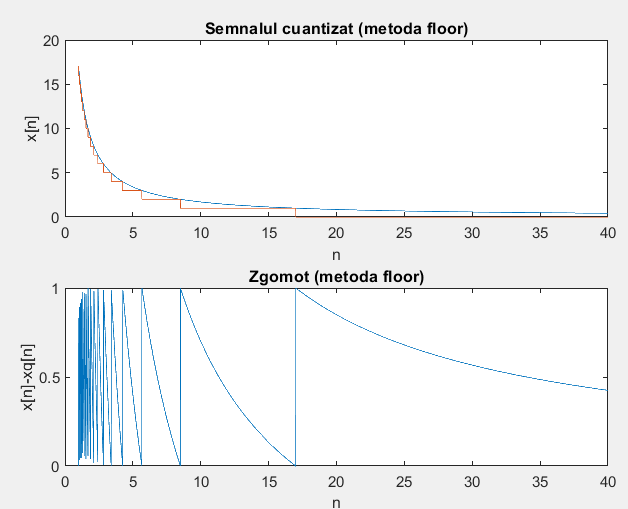
xlabel('n'); ylabel('x[n]'); title('Semnalul cuantizat (metoda floor)');

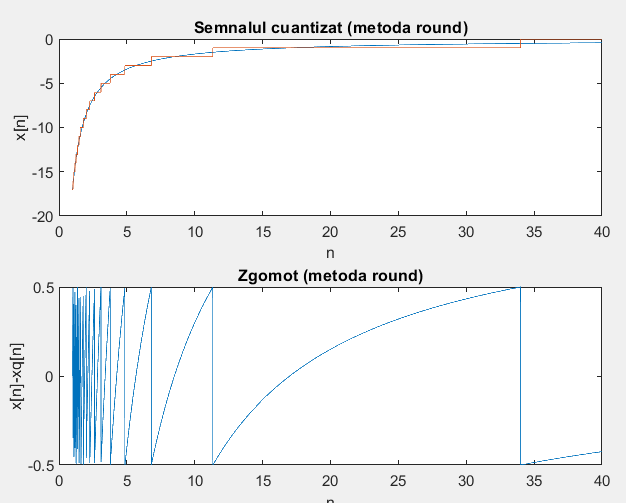
z=y-y1;

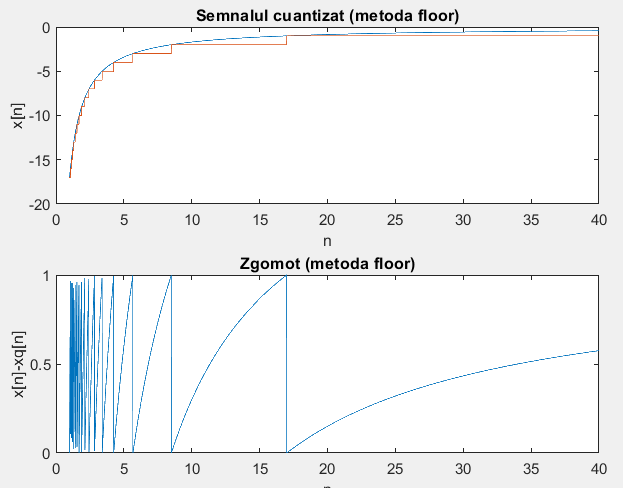
subplot(2,1,2), plot(n,z);

xlabel('n'); ylabel('x[n]-xq[n]'); title('Zgomot (metoda floor)');









Ex 2.3: Să se genereze 15 eșantioane de zgomot alb gaussian cu media zero și dispersia 0,2.

Indiciu: pentru a genera zgomot alb gaussian se poate folosi funcția rand:

z = 0.2\*randn(1,15)

Să se cuantizeze semnalul z folosind un cuantizor uniform cu nivelurile {0, ± 1/4 , ± 2/4 , ± 3/4 , ±1}.

z = 0.2\*randn(1,15);

plot(z);

title('Semnalul original');

figure(2)

q=0.25;

zq = round(z/q)\*q;

subplot(2,2,1), plot(zq);

title('Semnalul cuantizat (q=+-1/4)');

q=0.5;

zq = round(z/q)\*q;

subplot(2,2,2), plot(zq);

title('Semnalul cuantizat (q=+-2/4)');

q=0.75;

zq = round(z/q)\*q;

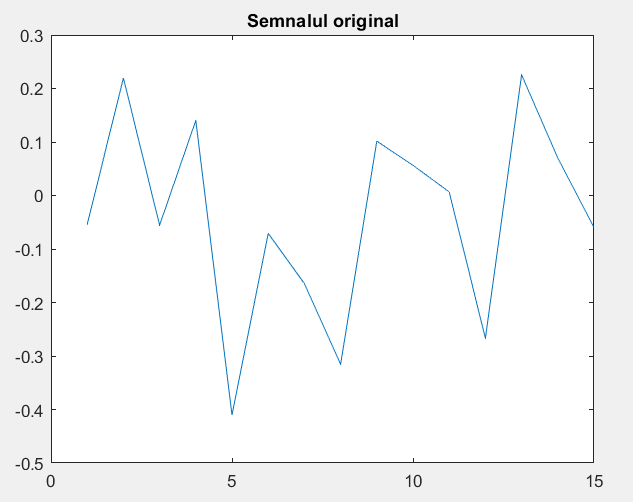
subplot(2,2,3), plot(zq);

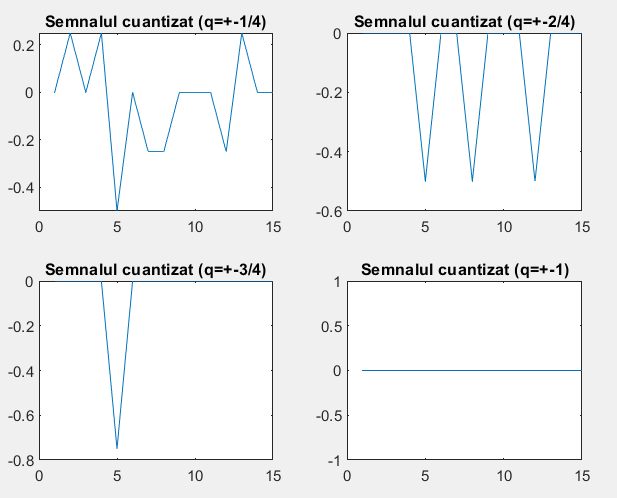
title('Semnalul cuantizat (q=+-3/4)');

zq = round(z);

subplot(2,2,4), plot(zq);

title('Semnalul cuantizat (q=+-1)');





Ex 2.4: Cuantizați semnalul din Exercițiul 4.1.1 pe 8 respectiv, 16 biți. Reprezentați semnalul cuantizat alături de zgomotul de cuantizare.

fs=8000;

t = 0 : 1/fs : 1024/fs;

x = 0.5\*sin(2\*pi\*50\*t)+0.2\*sin(2\*pi\*230\*t+pi/3);

figure(1)

N=256; %8 biti

x1 = floor((N-1)\*x)/(N-1);

subplot(2,1,1), plot(t,x,t,x1);

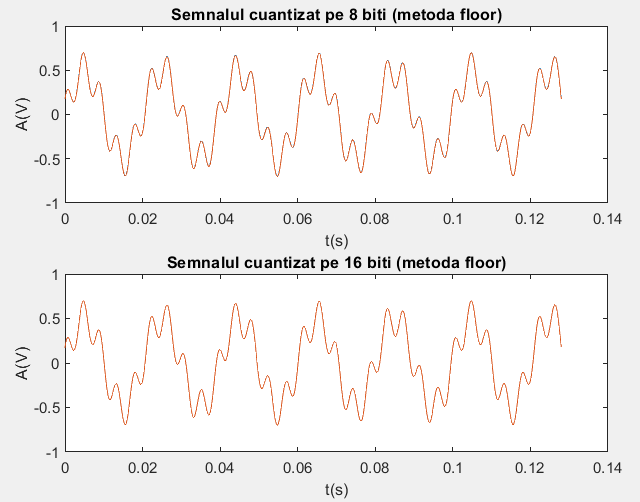
xlabel('t(s)'); ylabel('A(V)'); title('Semnalul cuantizat pe 8 biti (metoda floor)');

N=65536; %16 biti

x1 = floor((N-1)\*x)/(N-1);

subplot(2,1,2), plot(t,x,t,x1);

xlabel('t(s)'); ylabel('A(V)'); title('Semnalul cuantizat pe 16 biti (metoda floor)');



Ex 3:

a) Executarea calculelor cu valorile recomandate în antetul script-ului, reprezentarea semnalelor în formă originară şi după reconstrucție, precum şi a diferenţelor dintre ele, în

spaţii grafice diferite sau în acelaşi spaţiu.

b) Executarea de calcule cu valorile diferite pentru frecvenţele1f şi2f , eventual pentru amplitudini diferite ale sinusoidelor componente, utilizând pentru fiecare caz trei valori ale frecvenţei de eşantionare: mai mică, egală şi mai mare (de 10 ori) ca frecvenţa Nyquist.

c) Aprecierea vizuală şi/sau cantitativ a modificărilor semnalului reconstituit, calitatea reconstrucției semnalului în zona de început şi de final a intervalului de timp observat.

d) Studierea şi a altor semnale de bandă limitată, recomandate de profesor.

% optiune de reprezentare (rep=0) grafic unic, (rep>0) grafice separate

tmax=2; % timpul maxim de reprezentare

fi=pi/2; % faza componentei secundare

f1=8; % frecventa componentei primare

f2=13; % frecventa componentei secundare

% pasul utilizat la reprezentarea grafica

pasmic=0.001;

fes=4.1; % frecventa de esantionare

rep=1;

tes=1/fes; % perioada esantioanelor

a1=2.4;

a2=1.3;

t=0:pasmic:tmax;

% pregatirea graficului 1 (semnal original)

y=a1\*cos(2\*pi\*f1\*t)+a2\*cos(2\*pi\*f2\*t-fi);

t1=0:tes:tmax;

n=round(tmax/tes)+1;

% pregatirea graficului 2 (esantionarea)

y1=a1\*cos(2\*pi\*f1\*t1)+a2\*cos(2\*pi\*f2\*t1-fi);

% pregatirea graficului 3 (reconstituire)

y2=y1(1)\*sinc(t/tes);

for k=1:(n-1)

y2=y2+y1(k+1)\*sinc(t/tes-k);

end

if rep>0

subplot(4,1,1)

plot(t,y) % trasarea graficului 1 (semnal)

hold on

ylabel('Original')

title('ESANTIONAREA SEMNALELOR')

plot([0 tmax],[0 0])

for i=1:n

subplot(4,1,2)

% trasarea graficului 2 (esantionarea)

plot([t1(i) t1(i)],[0 y1(i)],'b:')

hold on

ylabel('Esantionat')

plot([0 tmax],[0 0])

end

subplot(4,1,3)

% trasarea graficului 3 (reconstituire)

plot(t,y2,'k')

hold on

plot([0 tmax],[0 0])

for i=1:n

plot([t1(i) t1(i)],[0 y1(i)],'b:')

end

ylabel('Reconstituire')

subplot(4,1,4)

% trasarea graficului 4 (diferente)

plot(t,y-y2,'r')

hold on

plot([0 tmax],[0 0])

ylabel('Diferente')

xlabel('Timp(s)')

else

plot(t,y) % trasarea graficului 1 (semnal)

hold on

plot([0 tmax],[0 0])

for i=1:n

% trasarea graficului 3 (reconstituire)

plot([t1(i) t1(i)],[0 y1(i)],'b:')

end

plot(t,y2,'k')

% trasarea graficului 4 (diferente)

plot(t,y-y2,'r')

plot([0 tmax],[0 0])

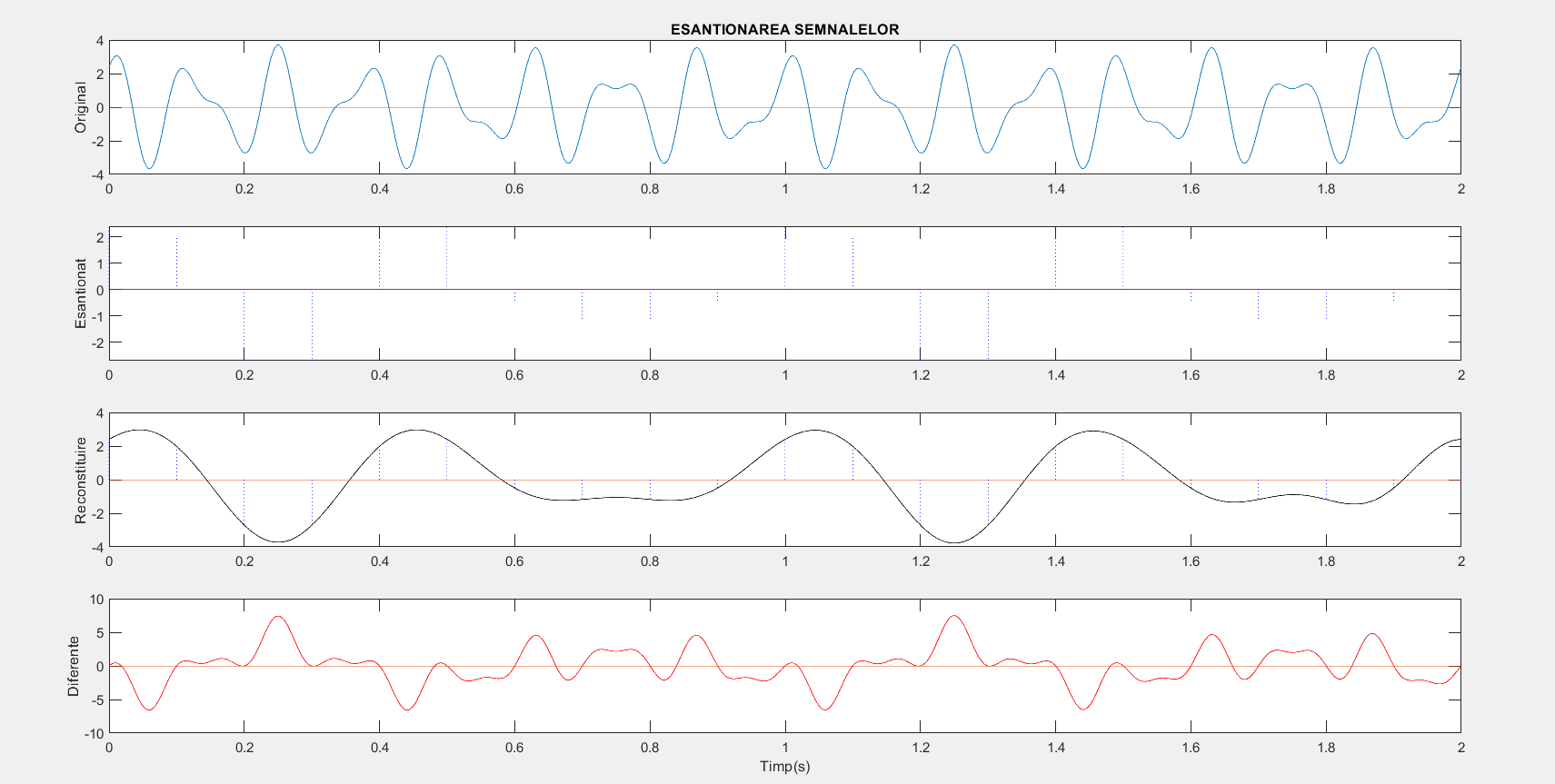
title('ESANTIONAREA SEMNALELOR')

ylabel('Semnal(albastru)/Reconstituire(negru)/Diferente(rosu)')

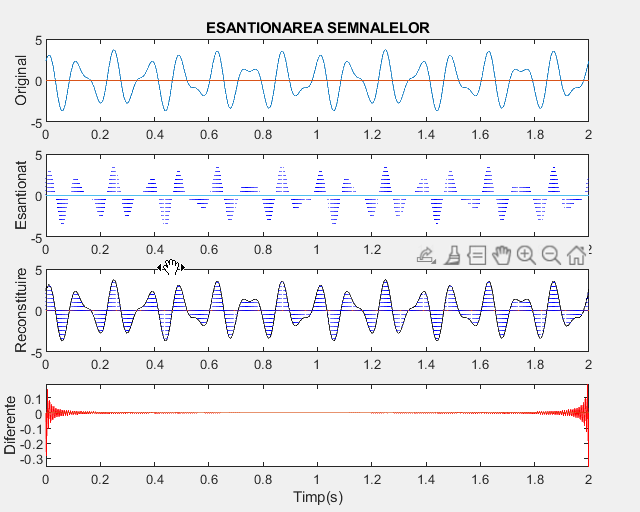
xlabel('Timp(s)')

end

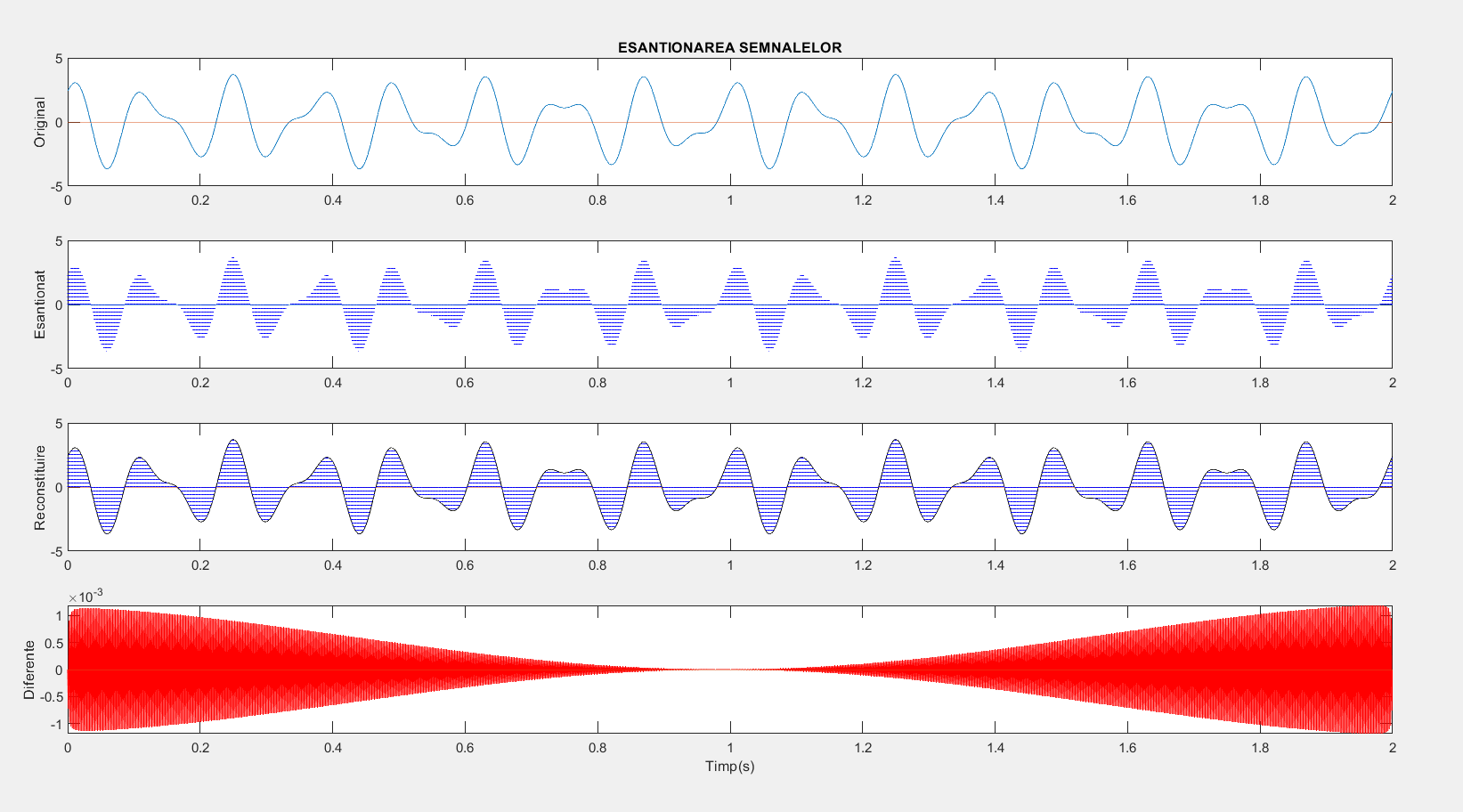
Fes = 10



Fes = 260



Fes = 999



**Concluzie**: În cadrul acestei lucrări de laborator, am explorat conceptele de eșantionare, cuantizare, zgomot de cuantizare și interpolare a semnalelor eșantionate. Am înțeles că eșantionarea implică procesul de măsurare a unui semnal continuu și discretizarea acestuia în puncte discrete de timp. Cuantizarea, pe de altă parte, presupune definirea semnalului continuu doar în anumite momente de timp, prin asignarea unor nivele sau valori discrete. În timpul cuantizării, am înțeles că apar erori de cuantizare care se datorează discreției valorilor cuantificate. Aceste erori pot duce la pierderi de informație și pot introduce distorsiuni în semnalul reconstruit. Zgomotul de cuantizare reprezintă fluctuațiile și erorile adiționale introduse în semnalul cuantizat din cauza procesului de cuantizare. Pentru a reconstrui un semnal cuantizat și a reduce efectele erorilor de cuantizare, am învățat despre interpolare. Interpolarea implică estimarea valorilor semnalului între punctele de eșantionare, folosind diferite metode de reconstrucție, precum interpolarea liniară sau cea polinomială. În concluzie, această lucrare de laborator ne-a ajutat să înțelegem procesul de eșantionare, cuantizare, zgomot de cuantizare și interpolare a semnalelor. Am realizat că eșantionarea și cuantizarea sunt procese esențiale în prelucrarea semnalelor și telecomunicații, dar pot introduce erori și distorsiuni. Utilizarea corectă a tehnicilor de interpolare poate contribui la reconstruirea semnalului original cu o precizie mai mare și la reducerea efectelor erorilor de cuantizare. Aceste cunoștințe ne vor fi utile în domenii precum prelucrarea semnalelor digitale, înregistrarea audio și video, precum și în teoria informației.