Subiectul nr. 2

Răspunsul în frecvență al sistemelor liniare invariante în timp (SLIT)

Reprezentarea de stare (în domeniul timp) a SLIT

(1)

unde:

,

A, B, C, D sunt matrice constante

Reprezentarea complexă a SLIT (prin funcție de transfer):

La TS (Teoria Sistemelor) s-a arătat că dacă:

atunci, dacă x(0) = 0:

Se observă cum mărimea:

descrie modificarea unei mărimi sinusoidale ca amplitudine și ca fază atunci când pulsația ω variază:

Programul Matlab care implementează teoria de mai sus este:

%sys =10000/[(s+10)\*(s+20)\*(s+50)]

sys=zpk([],[-10,-20,-50],1e4);

% extragem numaratorul si numitorul

[num,den]=tfdata(sys,'v');

% generam 100 de pulsatii spatiate logaritmic intre 10^-1 si 10^3

omega=logspace(-1,3,100);

% se calculeaza H(j\*omega)= num(j\*omega)/den(j\*omega)

H=polyval(num,j\*omega)./polyval(den,j\*omega);

% se calculeaza amlitudinea si faza

Amplitudine=abs(H);

Faza=180/pi\*phase(H);

% se afiseaza rezultatele

subplot(211);

plot(log10(omega),20\*log10(Amplitudine));grid;

title('Caracteristica Amplitudine-pulsatie')

xlabel('log10(omega)')

ylabel('20\*log10(Amplitudine) [dB]')

%semilogx(omega,20\*log10(A))

subplot(212);

plot(log10(omega),Faza);grid;

title('Caracteristica Faza-pulsatie')

xlabel('log10(omega)')

ylabel('Faza [grade]')

Aceeași problemă este realizată și de funcția Matlab „bodeplot”, cu diferența că aceasta este capabilă să adapteze gama de reprezentare a pulsației (la scară logaritmică) în funcție de dinamica sistemului.

In continuare vom prezenta o modalitate prin care vom putea implementa un comportament asemănător și pentru programul implementat de noi.

Vom implementa o funcție, numită „gama\_frecvente” prin care, furnizând polii funcției de transfer, ni se vor returna puterile lui 10 între care se vor calcula caracteristicile de frecvență.

function [p\_min,p\_max] = gama\_frecvente(poli)

pol\_min=min(abs(real(poli)));

pol\_max=max(abs(real(poli)));

p=0;

while (10^p >= pol\_min)

p=p-1;

p\_min=p;

end

p=0;

while (10^p < pol\_min)

p=p+1;

p\_min=p;

end

p\_min=p\_min-1;

p=0;

while (10^p<pol\_max)

p=p+1;

p\_max=p;

end

p=0;

while (10^p>=pol\_max)

p=p-1;

p\_max=p;

end

p\_max=p\_max+1;

end

Programul principal ar putea fi:

%sys =10000/[(s+10)\*(s+20)\*(s+50)]

sys=zpk([],[-10,-20,-50],1e4);

% extragem numaratorul si numitorul

[num,den]=tfdata(sys,'v');

[k\_min,k\_max]=gama\_frecvente(roots(den));

omega=logspace(k\_min,k\_max,100);

H=polyval(num,j\*omega)./polyval(den,j\*omega);

Amplitudine=abs(H);

Faza=180/pi\*phase(H);

subplot(211);

plot(log10(omega),20\*log10(Amplitudine));grid;

title('Caracteristica Amplitudine-pulsatie')

xlabel('log10(omega)')

ylabel('20\*log10(Amplitudine) [dB]')

%semilogx(omega,20\*log10(A))

subplot(212);

plot(log10(omega),Faza);grid;

title('Caracteristica Faza-pulsatie')

xlabel('log10(omega)')

ylabel('Faza [grade]')

Studiul în frecvență al SLIT nu este relevant decât dacă este pus în legătură cu răspunsul SLIT la intrări de tip sinusoidal. Aici de mare ajutor este Simulink.

Să analizăm răspunsul în frecvență al SLIT de mai sus:

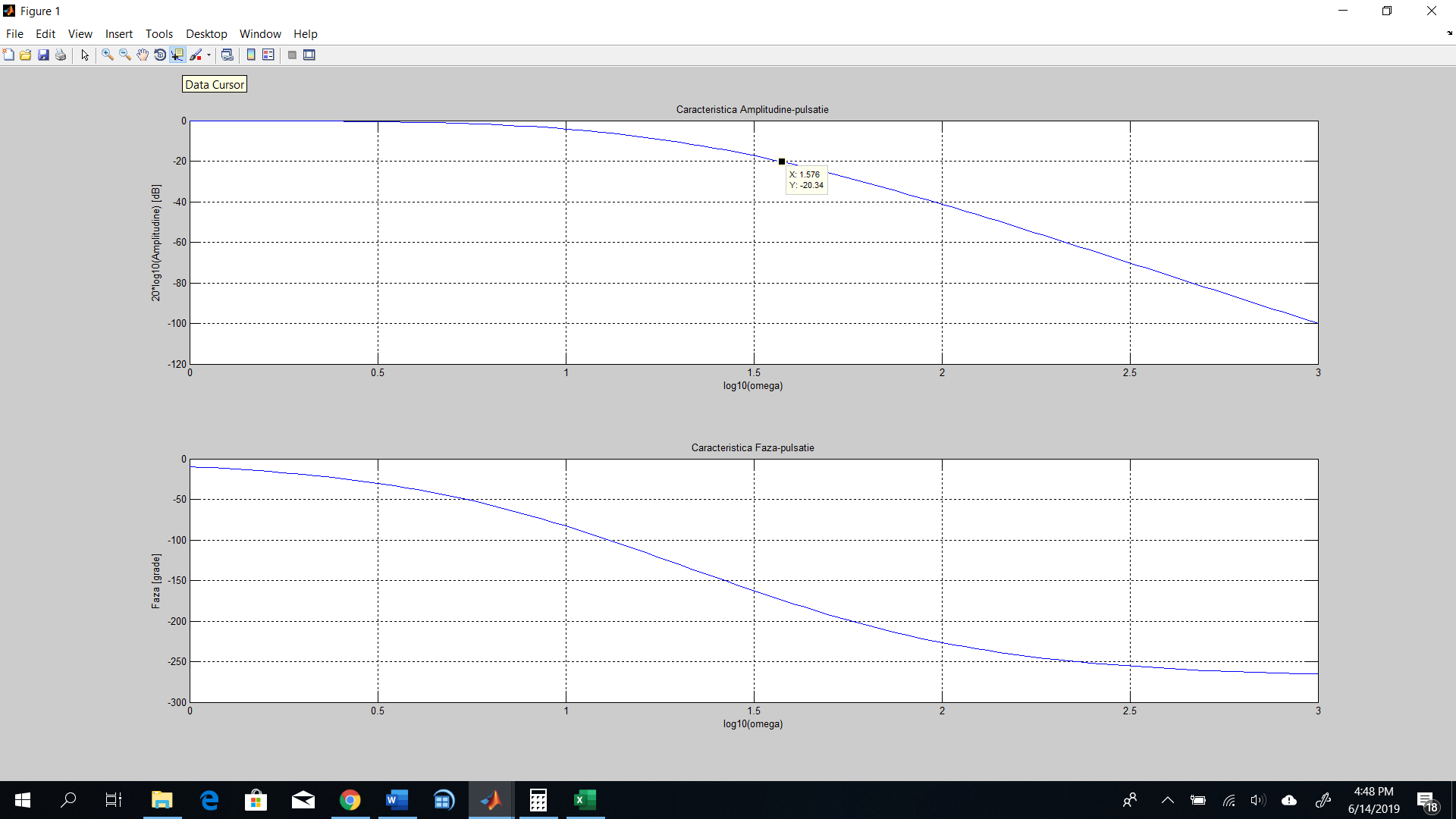
* Cu ajutorul „Data cursor” putem vizualiza coordonatele unor puncte de pe grafic.
* Spre exemplu, pentru X = 1.576 vom avea Y = -20.34

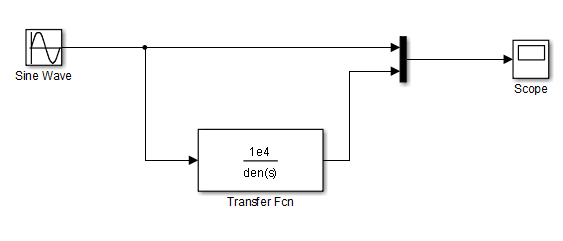
Aceste rezultate se interpretează astfel:

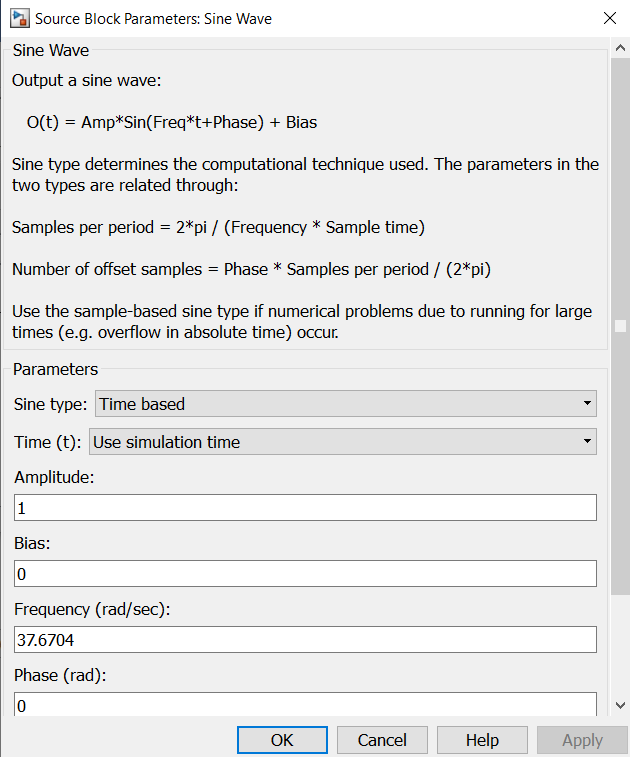
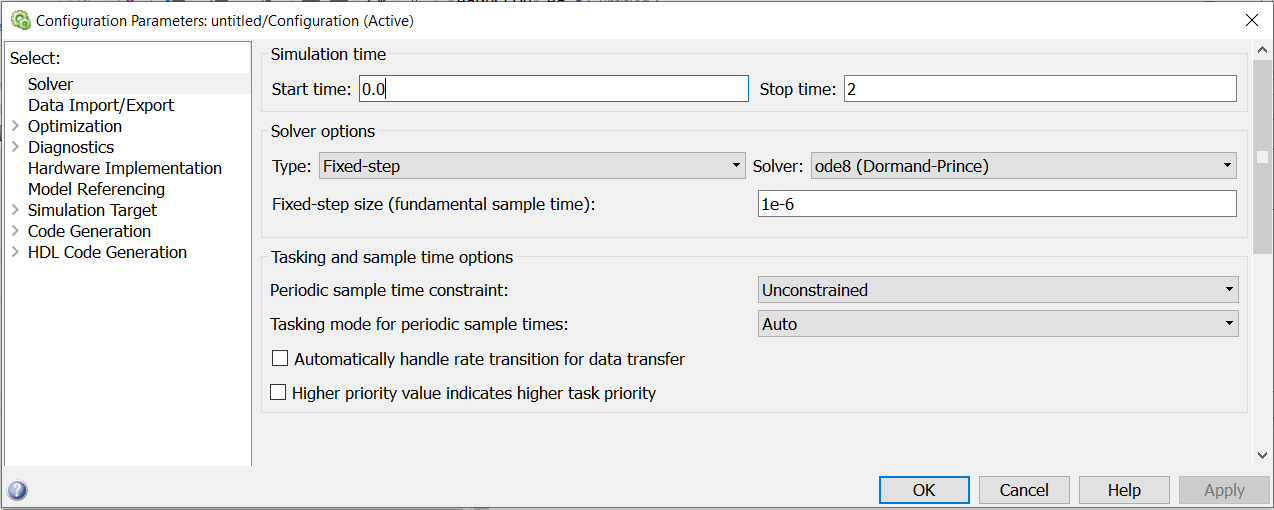
X = 1.576 => lg(ω) = 1.576 => ω = 101.576 => ω = 37.6704 [rad/sec]

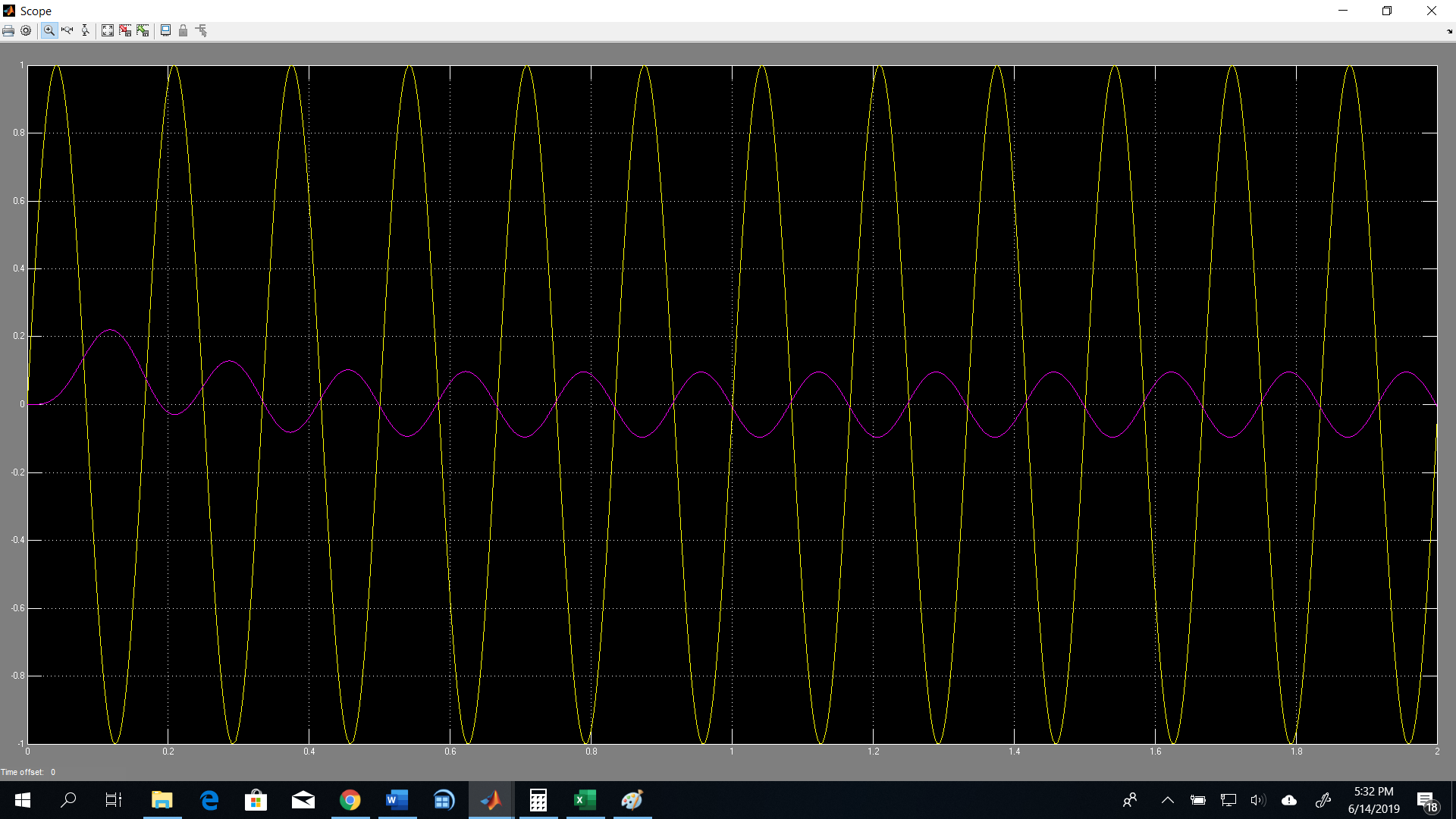
Y = 20\*lg(A(ω)) = -20.34 => A(ω) = 10-20.34/20 = 0.0962 ~= 0.1

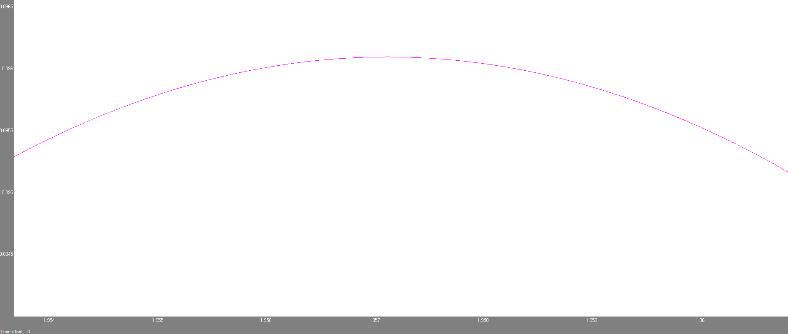
Din interpretarea datelor, concluzionăm ca, dacă la intrarea sistemului aplicăm un semnal sinusoidal cu amplitudine unitară și pulsație de 37.6704 [rad/sec], la ieșire vom obține un semnal defazat cu o amplitudine de 0.0962









Se observă prin simulare în timp corectitudinea rezultatelor.

Probleme de examen:

1. Explicați cum au fost implementate programele de mai sus
2. Faceți un program în Matlab care să simuleze răspunsul sistemului la o intrare de tip sinusoidă de o anumită pulsație și verificați rezultatele experimentale obținute