Laboratory 4

Designing automatic systems with MATLAB / Simulink

1.Purpose of the exercise:

- learning methods of creating linear automatic systems models & transforming model forms
- creating block diagrams of automatic systems
- drawing time responses and frequency responses

2.Theoretical introduction

<u>2.1. Automatic systems models</u>

MATLAB often uses two types of linear dynamic models:

- state-space and output equation

To achieve a complete definition of the model, you should define matrices A, B, C and D. For instance:

$$A = \begin{bmatrix} -3 & -2 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 2 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \end{bmatrix}$$

Using ***ss(A, B, C, D)*** we get model presentation in *Command Window*.

- matrix transfer function (for SIMO systems only – Single-Input Multi-Output).

To define a transfer function you should input two vectors, containing coefficients of numerator and denominator. Coefficients are put in descending order of operator **s** powers. For instance: vectors NUM=[1 2], DEN=[1 3 2] give the transfer function shown below:

$$G(s) = \frac{s+2}{s^2+3s+2}$$

Using ***printsys(NUM,DEN)*** command, the transfer function is printed on the screen.

<u>2.2. Transforming model forms</u>

- ***ss2tf*** and ***tf2ss*** functions

Syntax :

[NUM, DEN] = *ss2tf*(A, B, C, D, Ui) – **s**tate-**s**pace to **t**ransfer **f**unction conversion
[A, B, C, D] = *tf2ss*(NUM, DEN) – **t**ransfer **f**unction to **s**tate-**s**pace conversion

A – state (system) matrix, B – input (control) matrix, C – output matrix, D – feedthrough matrix, Ui – input number (for multi-input systems only).

For more information use ***help ss2tf*** or ***help tf2ss*** in MATLAB.

2.3. Block diagrams modeling

Several functions that allow to receive resultant models for systems with feedback, serial and parallel connections, are shown below:

- ***feedback*** - feedback connection of the two models
  [NUM, DEN]= *feedback*(NUM1, DEN1, NUM2, DEN2, SIGN)

- ***series*** - series connection of the two models
  [NUM, DEN]=*series*(NUM1, DEN1, NUM2, DEN2)

- ***paralell*** - parallel connection of the two models
  [NUM, DEN]=*paralell*(NUM1, DEN1, NUM2, DEN2)

If SIGN = 1 then positive feedback is used. If SIGN = –1 or SIGN is omitted, negative feedback is used. For more information use HELP in MATLAB.

2.4. Time response determination

Unit impulse response of a linear system is calculated by ***impulse***. The impulse response is a response to a Dirac input for continuous-time systems. Syntax:

- *impulse*(A,B,C,D,Ui)
- *impulse*(NUM, DEN)
- [Y,X,T]=*impulse*(A,B,C,D,Ui)
- [Y,X,T]=*impulse*(NUM, DEN)

step calculates the unit step response of a linear system. Syntax:

- *step*(A,B,C,D,Ui)
- *step*(NUM, DEN)
- [Y,X,T]= *step*(A,B,C,D,Ui)
- [Y,X,T]= *step*(NUM, DEN)

2.5 Frequency response determination

nyquist calculates the Nyquist frequency response of the model. Syntax:

- *nyquist*(A,B,C,D,Ui)
- *nyquist*(NUM, DEN)
- [re,im,w]=*nyquist*(NUM, DEN)

bode computes the magnitude and phase of the frequency response. Syntax:

- *bode*(A,B,C,D,Ui)
- *bode*(NUM, DEN)
- [amplitude,phase,w]=bode(NUM, DEN)

3. Procedure of the laboratory

3.1. For matrices given below, convert state-space model to a transfer function

a) $\mathbf{A} = \begin{bmatrix} -4, & 2 \\ 2, & -1 \end{bmatrix}$ $\mathbf{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ $\mathbf{C} = \begin{bmatrix} 0 & 0 \end{bmatrix}$ $\mathbf{D} = [0]$

b) $\mathbf{A} = \begin{bmatrix} -1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -3 & 0 \end{bmatrix}$ $\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ $\mathbf{C} = \begin{bmatrix} 1 & 1 & 0 \end{bmatrix}$ $\mathbf{D} = [0]$

3.2. For transfer functions given below, convert models to state-space

a) $G(s) = \frac{4s}{2s+1}$, b) $G(s) = \frac{2}{s^2+4s+6}$, c) $G(s) = \frac{1}{5s}$, d) $G(s) = 3$

3.3. Determine time and frequency responses for the following automatics elements:

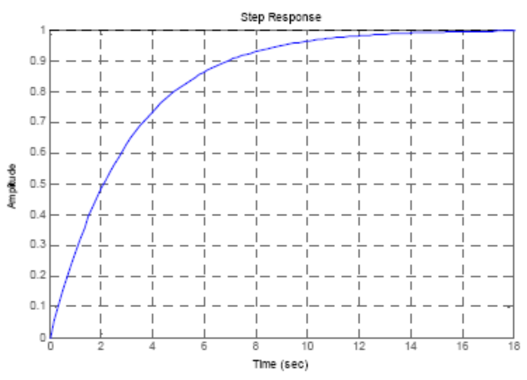
a) proportional element: K=2;
b) ideal integral element: K=3;
c) ideal differential element: K=5;
d) real differential element (with 1st order inertia): K=0.1 , T=8;
e) 1st order inertial element: K=3 , T=1;
f) 2nd order inertial element: K=2 , T_1=2, T_2=4;
g) 2nd order oscillatory element: K=1 , ω=1, ζ=0.2;
h) 2nd order oscillatory element: K=1 , ω=2, ζ=0.2;

Example 1

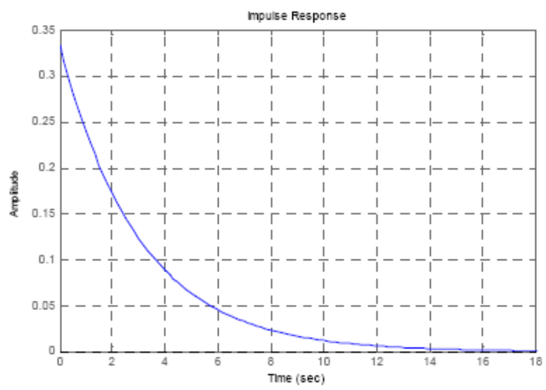
Determine time and frequency responses for 1st order inertial element:

$$G(s) = \frac{K}{Ts+1}$$ K = 1, T = 3

```
>> l=[1];
>> m=[3,1];
>> step(l,m)
>> grid
```

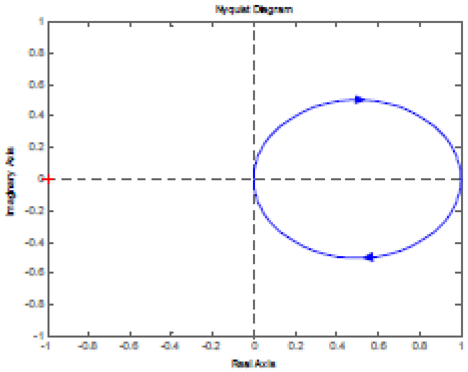


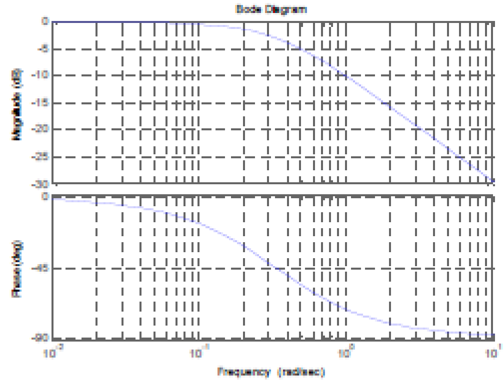

```
>> impulse(l,m)
>> grid
```



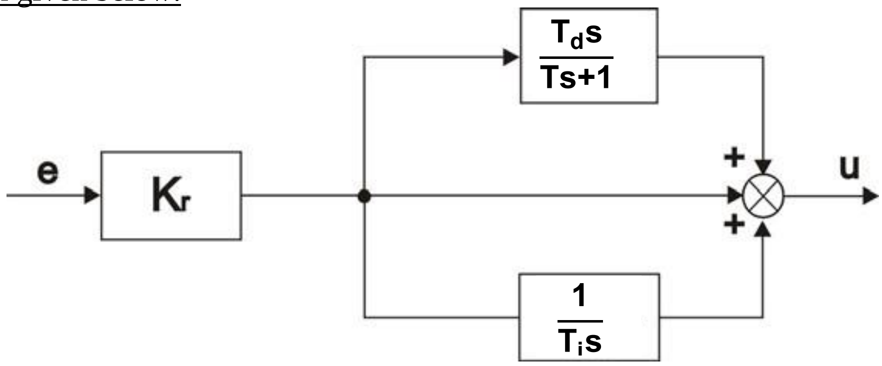
```
>> nyquist(l,m)
```

```
>> bode(l,m)
>> grid
```



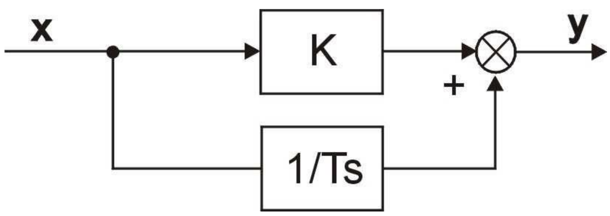


<u>3.4. Assuming: Kr=1.5 ; T_d =3 ; T_i =2 ; T=1, write the m-file that plots step and impulse responses, Nyquist frequency response, and Bode magnitude and phase frequency responses for a system given below.</u>



Example 2

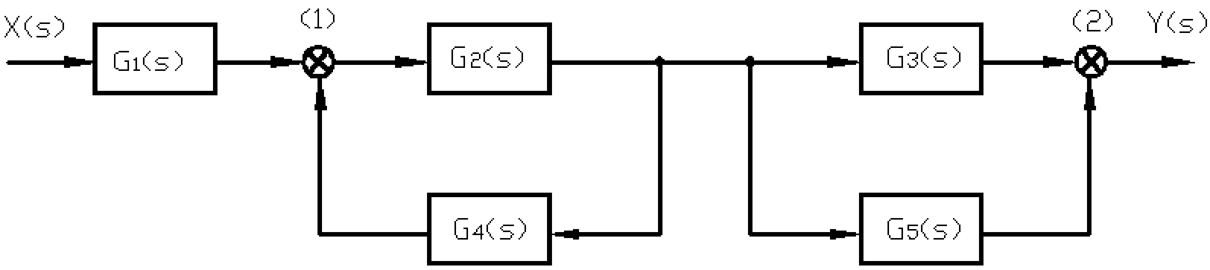
Assuming: K=2 ; T =4, write the m-file that plots Nyquist frequency response, and Bode magnitude and phase frequency responses for a system given below.




```
%Data
    k=1.5;T=3;

%Numerator and denominator of: K+1/Ts
    [NUM, DEN]=parallel([k],[1],[1],[T 0]); w=0:0.01:200;
    [mod,phase,w]=bode(NUM, DEN);

% magnitude and phase of the frequency response
    nyquist(NUM,DEN,w);axis([-1 5 -5 2]);grid;pause

% logarithmic magnitude characteristics
    semilogx(w,20*log10(mod)); grid;ylabel('Lm [dB]');pause

% logarithmic phase characteristics
    semilogx(w,phase); grid;ylabel('phase [degrees]');pause;
```

3.5. For the system below:

(a) write the MATLAB m-file to calculate a resultant transfer function; present the results with ***printsys***,
(b) draw the step response of the system using ***step*** MATLAB command,
(c) build a Simulink model of the system with *Step* input block, and compare its output with the result obtained with ***step*** MATLAB command,
(d) ADDITIONAL: calculate the resultant transfer function analytically and compare it with the transfer function (a) calculated in MATLAB (enclose the solution).



PC no.	**1 / 8 / 15**	**2 / 9**	**3 / 10**	**4 / 11**	**5 / 12**	**6 / 13**	**7 / 14**
$G_1(s)$	$\frac{2}{s+1}$	$\frac{4}{2s+1}$	$\frac{1}{4s+1}$	$\frac{1}{3s+2}$	10	$\frac{4}{2s+1}$	$\frac{2}{s+1}$
$G_2(s)$	$\frac{4}{2s+1}$	10	$\frac{2}{s+1}$	$\frac{1}{4s+1}$	$\frac{1}{3s+2}$	$\frac{1}{4s+1}$	10
$G_3(s)$	10	$\frac{1}{4s+1}$	$\frac{1}{3s+2}$	$\frac{4}{2s+1}$	$\frac{2}{s+1}$	10	$\frac{4}{2s+1}$
$G_4(s)$	$\frac{1}{3s}$	$\frac{-2}{s}$	$\frac{4}{2s+1}$	-10	$\frac{-1}{4s+1}$	$\frac{2}{s}$	$\frac{1}{3s}$
$G_5(s)$	$\frac{1}{4s+1}$	$\frac{1}{3s+2}$	10	$\frac{2}{s+1}$	$\frac{4}{2s+1}$	$\frac{1}{3s+2}$	$\frac{1}{4s+1}$
(1)	−		−			−	−
(2)		−		−		−	−

3.6. Observe the influence of ξ, k, ω_o on the logarithmic plots of 2nd order oscillatory element:

$$G(s) = \frac{k\omega_o^2}{s^2 + 2\xi\omega_o s + \omega_o^2}$$

Write the m-file that allows to draw logarithmic plots on the assumption that:

(a) ξ = var , k, ω_o = const
(b) k = var , ξ, ω_o = const
(c) ω_o = var , k, ξ = const

Use instructions like *input*, *pause*, *hold on*, …

Task assignment:

PC no.		Observe the influence of	
7 / 14	4 / 11	ξ	k
2 / 9	5 / 12	k	ω_o
3 / 10	6 / 13	ξ	ω_o
1 / 8 / 15		ξ	k

Matching parameters:

$\xi_1 = 0.2$	$k_1 = 1.5$	$\omega_1 = 1.0$
$\xi_2 = 0.5$	$k_2 = 4.5$	$\omega_2 = 2.5$
$\xi_3 = 0.8$	$k_3 = 7.5$	$\omega_3 = 4.5$
$\xi_4 = 0.3$	$k_4 = 2.0$	$\omega_4 = 2.5$
$\xi_5 = 0.6$	$k_5 = 5.5$	$\omega_5 = 5.0$
$\xi_6 = 0.9$	$k_6 = 8.0$	$\omega_6 = 7.5$

Determine time and frequency responses and formulate comments about parameter influence.

References:

[1] G.F. Franklin, J.D. Powell, E. Emami-Naeini “Feedback control of dynamic systems”, Prentice Hall, New York, 2006.
[2] K. Ogata “Modern control engineering”, Prentice Hall, New York, 1997.
[3] R.H. Cannon “Dynamics of physical systems”, Mc-Graw Hill, 1967 (available in Polish as: R.H. Cannon “Dynamika układów fizycznych”, WNT, Warszawa, 1973).
[4] J. Kowal “Podstawy automatyki”, v.1 and 2, UWND, Kraków, 2006, 2007 (in Polish).
[5] W. Pełczewski “Teoria sterowania”, WNT, Warszawa, 1980 (in Polish).
[6] Brzózka J., Ćwiczenia z Automatyki w MATLABIE i Simulinku, Wydawnictwo Mikon, Warszawa 1997 (in Polish).
[7] Zalewski A., Cegieła R., MATLAB: obliczenia numeryczne i ich zastosowania, Wydawnictwo Nakom, Poznań 1996 (in Polish).
[8] MATLAB/Simulink documentation: http://www.mathworks.com/help/