

## **Chapter #4**

**EEE8013**

### **Linear Controller Design and State Space Analysis**

#### **Design of control system in state space using Matlab**

<b>1. Controllability and Observability.....</b>	<b>2</b>
<b>2. State Feedback Control.....</b>	<b>5</b>
<b>3. Linear Quadratic Regulator (LQR) .....</b>	<b>19</b>
<b>4. Open loop and closed loop estimators .....</b>	<b>22</b>

## 1. Controllability and Observability (Review)

A state space system is completely observable if the rank of the

observability matrix ( $M_o$ ),  $M_o = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$ , is  $n$  ( $n$  is the system order). This

implies that this matrix contains  $n$  linearly independent row or column vectors. (that is, the matrix  $M_o$  is non-singular having non-zero determinant).

If the rank of this matrix  $M_o$  is less than  $n$  then the system is unobservable.

To check if the system is observable or not in Matlab we use the command `rank(observ(A,C))`. The answer is the rank of the observability matrix  $M_o$  which should be  $n$  for the system to be observable.

### Example 4.1:

Check if the following state space system  $\left. \begin{array}{l} \dot{x} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 1 \end{bmatrix} u \\ y = \begin{bmatrix} 3 & 0 \end{bmatrix} x \end{array} \right\}$  is

observable. As in the lecture notes we can check the system observability by checking the determinant of the observability matrix  $M_o$  as follow:

$M_o = \begin{bmatrix} C \\ CA \end{bmatrix} = \begin{bmatrix} 3 & 0 \\ -6 & 0 \end{bmatrix} \Rightarrow |M_o| = \begin{vmatrix} 3 & 0 \\ -6 & 0 \end{vmatrix} = 0$ . Which means that the matrix

$M_o$  is singular and the system is unobservable. If we want to use Matlab:

```
>> A=[-2 0;0 -1];
>> C=[3 0];
>> rank(observ(A,C))
ans =
    1
```

This means that the rank of the observability matrix is 1 and since the system order  $n=2$  then the system is unobservable. ■

A state space system is completely controllable if the rank of the controllability matrix ( $M_c$ ),  $M_c = [B \ AB \ A^2B \ \dots \ A^{n-1}B]$ , is  $n$  ( $n$  is the system order). This implies that this matrix contains  $n$  linearly independent row or column vectors. (that is, the matrix  $M_c$  is non-singular having non-zero determinant). If the rank of this matrix  $M_c$  is less than  $n$  then the system is uncontrollable. To check if the system is controllable or not in Matlab we use the command `rank(ctrb(A,B))`. The answer is the rank of the controllability matrix  $M_c$  which should be  $n$  for the system to be controllable.

**Example 4.2:**

Check if the following state space system 
$$\left. \begin{aligned} \dot{x} &= \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 2 \\ 0 \end{bmatrix} u \\ y &= \begin{bmatrix} 3 & 2 \end{bmatrix} x \end{aligned} \right\} \text{ is}$$

controllable. As in the lecture notes we can check the system controllability by checking the determinant of the controllability matrix  $M_c$  as follow:

$$M_c = [B \ AB] = \begin{bmatrix} 2 & -4 \\ 0 & 0 \end{bmatrix} \Rightarrow |M_c| = \begin{vmatrix} 2 & -4 \\ 0 & 0 \end{vmatrix} = 0. \text{ Which means that the}$$

matrix  $M_c$  is singular and the system is uncontrollable. If we want to use Matlab:

```
>> A=[-2 0;0 -1]; B=[2 0]';
```

```
>> rank(ctrb(A,B))
```

```
ans =
```

```
1
```

This means that the rank of the controllability matrix is 1 and since the system order  $n=2$  then the system is uncontrollable. ■

**Example 4.3:**

Show that the following system  $\dot{x} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix}x + \begin{bmatrix} 2 \\ 1 \end{bmatrix}u$ ,  $y = \begin{bmatrix} 3 & 2 \end{bmatrix}x$  is

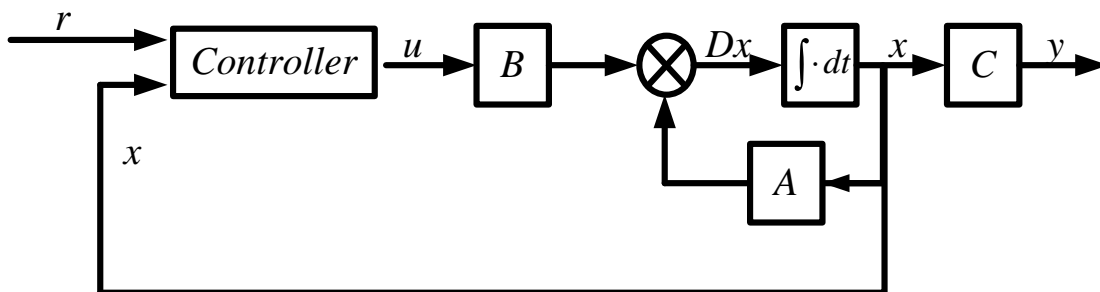
minimal realisation.

```
>> clear all
>> A=[-2 0;0 -1];B=[2 1]';C=[3 2];
>> rank(ctrb(A,B))
ans =
     2
>> rank(observ(A,C))
ans =
     2
```

■

**2. State Feedback Control**

In state space systems we can have two kinds of feedback; the output and the state feedback (we will only study the state feedback method):



The task of the controller is to produce the appropriate control signal  $u$  that will insure that  $y=r$ . Let's assume the simplest form of control:  $u(t) = K_1 R(t) - Kx(t)$ , where  $K_1$  and  $K$  are appropriately selected gain vectors/matrices. The exact dimensions of these vectors will be defined later.

The closed loop system state space model can be described by the following equations:

$$\left. \begin{array}{l} x(t) = A_{CL}x(t) + B_{CL}r(t) \\ y(t) = C_{CL}x(t) + D_{CL}u(t) \end{array} \right\} \text{where } \left. \begin{array}{l} A_{CL} = A - BK \\ B_{CL} = BK_1 \\ C_{CL} = C - DK \\ D_{CL} = DK_1 \end{array} \right\} \text{ are the closed loop matrices. If}$$

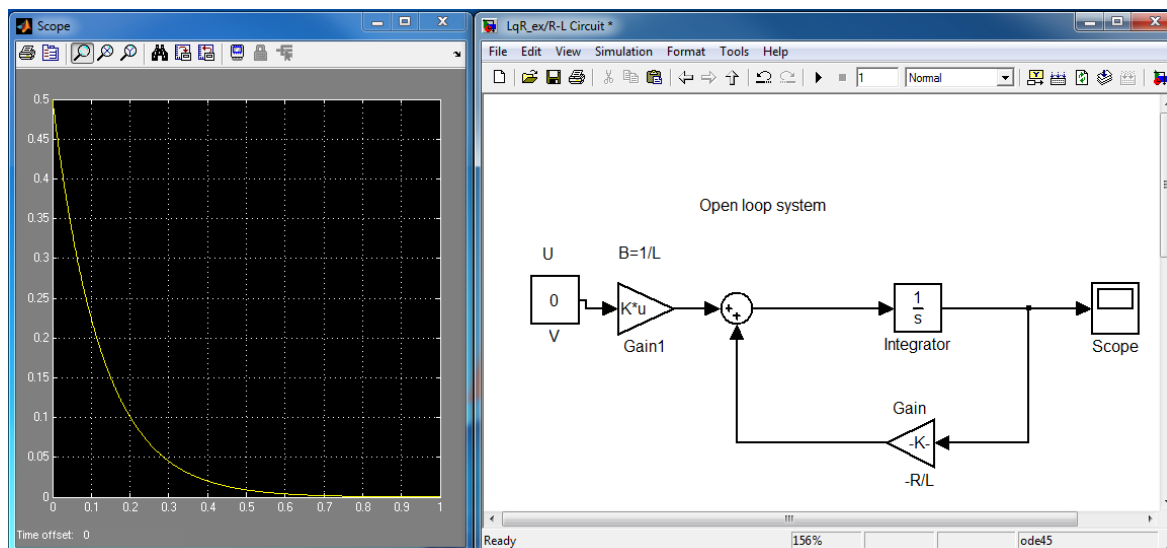
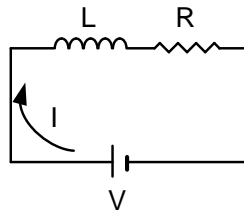
$r=0$  the system is called regulator.

The closed loop state matrix  $A_{CL}$  is a function of  $K$ , therefore by appropriate changing  $K$  we can change the eigenvalues of  $A_{CL}$  which means that we can improve the system performance, i.e. to make it faster/stable. This method is called **pole placement**.

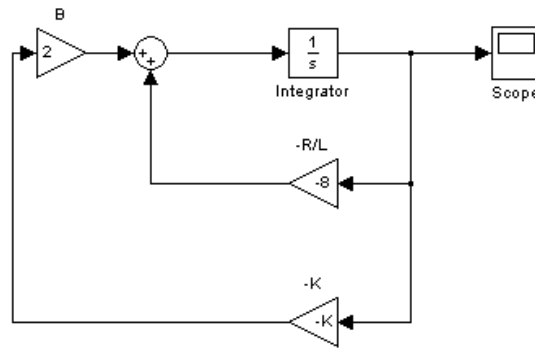
### WE MUST CHECK IF THE SYSTEM IS CONTROLLABLE.

#### Example 4.4:

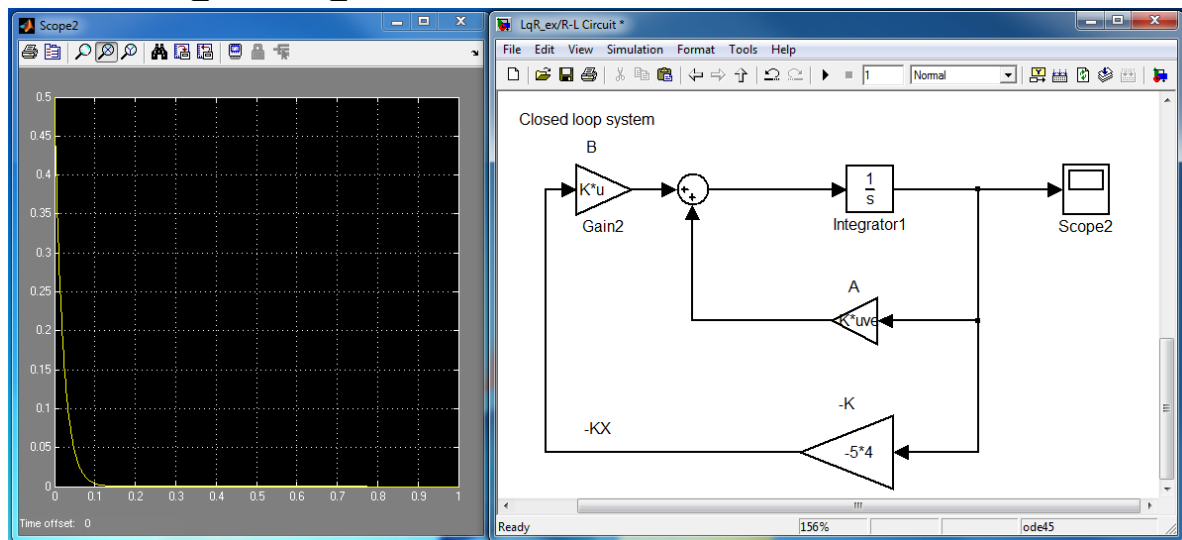
Assume that we have an RL circuit with  $R=4$  ohms and  $L=0.5\text{H}$  and  $x(0)=0.5$



If we want to make the system faster then we can use a state feedback control strategy. The new signal  $u$  is  $-Kx$

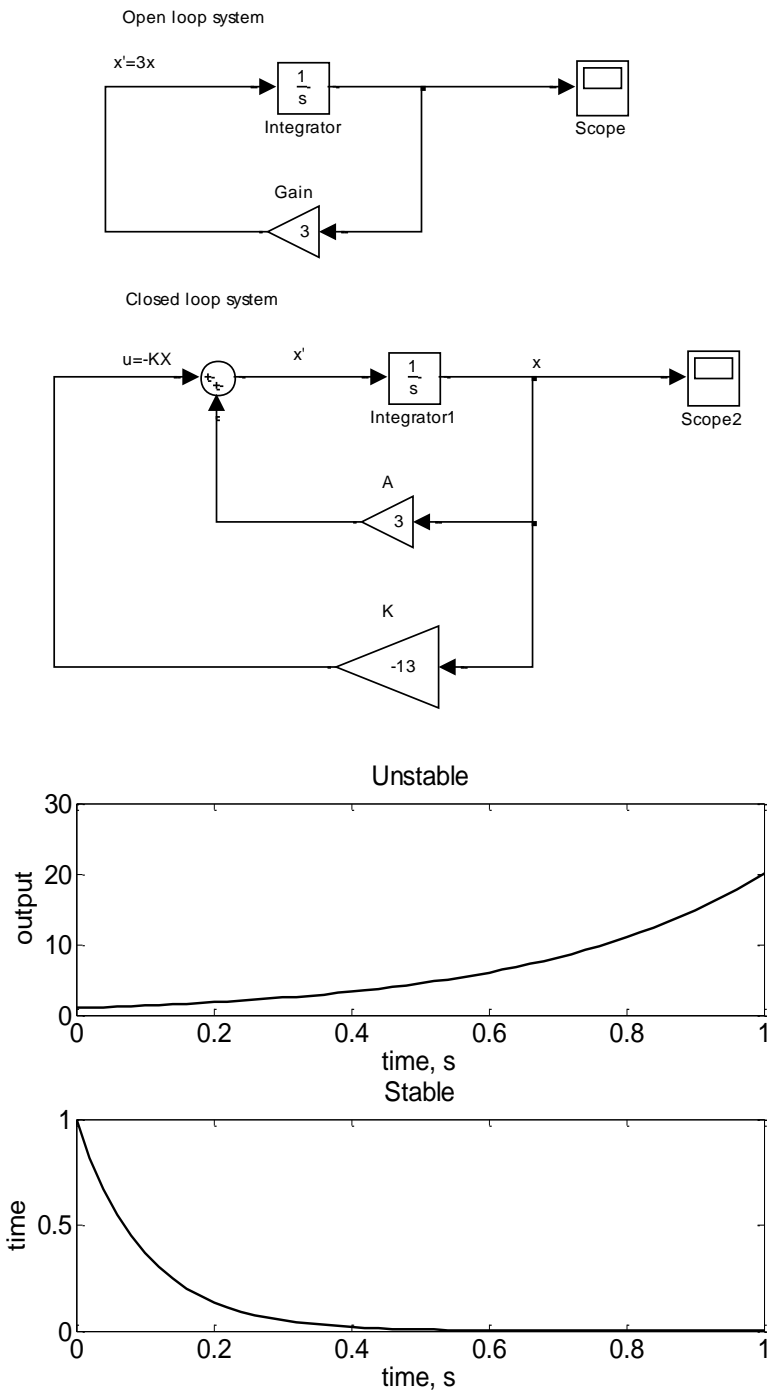


This implies that  $A_{CL} = A - BK = \frac{-R}{L} - \frac{K}{L} = \frac{-R-K}{L}$ . The new system has an eigenvalue at  $\frac{-R-K}{L}$  (EXACTLY as before). If we want to place the eigenvalue at  $-6R/L$  then  $\frac{-R-K}{L} = -6\frac{R}{L} \Rightarrow K = 5R$ :



**Example 4.5:**

Assume  $\dot{x} = 3x + u$  (unstable system),  $x(0)=1$ . We use a state feedback controller  $u = -kx$  which implies that the Closed Loop (CL) state equation is  $\dot{x} = (3-k)x$ , hence the eigenvalue of the CL system is  $3-k$ . If we want the CL eigenvalues at  $-10$  (stable and very fast) then  $k=13$ :



**Example 4.6:**

Find the eigenvalues of  $\left. \begin{array}{l} \dot{x}(t) = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u \\ y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t) \end{array} \right\}$ . If the system is unstable

create a controller that will stabilise the system. Find the response if  $\mathbf{x}(0) = [1 \ 1]^T$ .

```
>> clear all
>> A=[1 2;3 4];B=[1 0]';C=[1 0];
>> eig(A)
ans =
    -0.3723
     5.3723
```

The eigenvalues are: -0.3723, 5.3723. Hence we need to create a feedback controller to stabilise the system. Before that we need to check if the system is controllable??

```
>> rank(ctrb(A,B))
ans =
     2
```

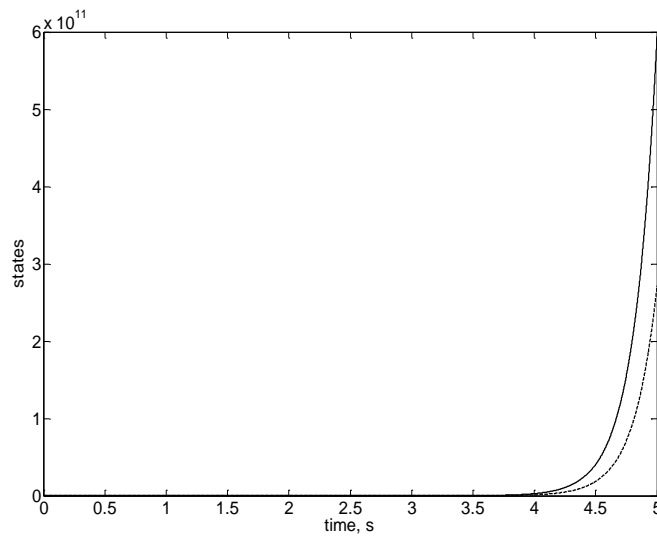
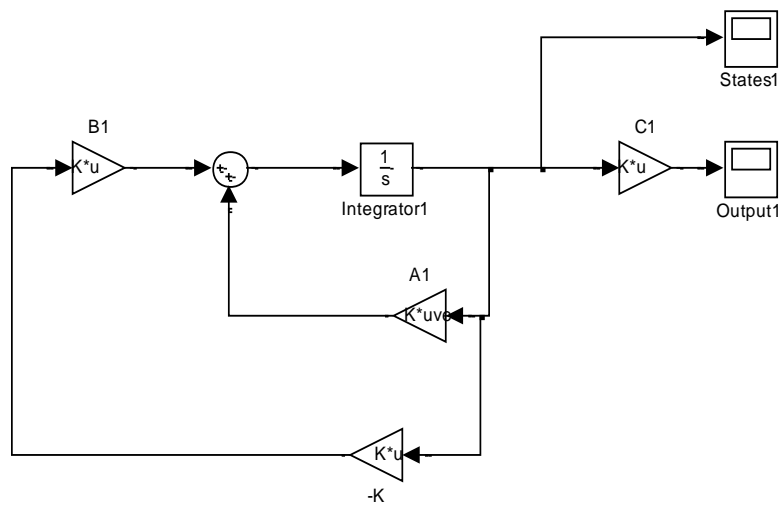
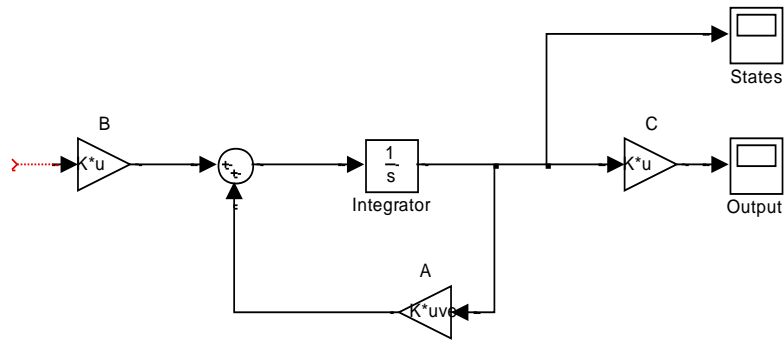
Hence the system is controllable and I can then use a state feedback controller using a pole placement strategy. Our task is to find the gain vector  $K$  that will be used in the state feedback control law to place the closed loop eigenvalues at  $\lambda_1$  and  $\lambda_2$ . In Matlab we can use the command  $K = \text{place}(A, B, [\lambda_1 \ \lambda_2])$  to find  $K$ .

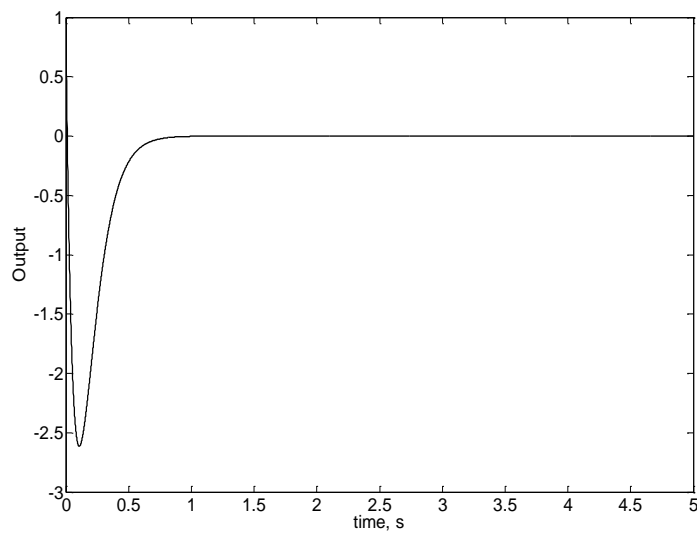
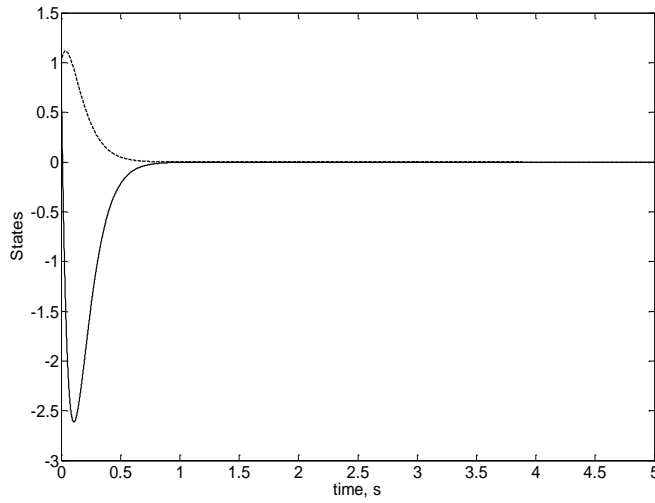
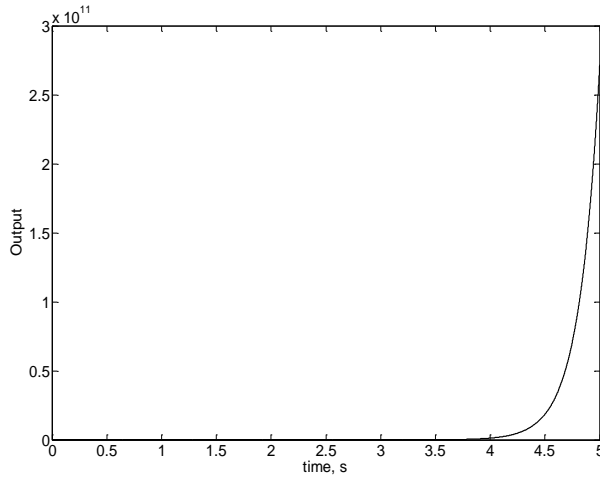
Assume that we want to place the closed loop poles at -10 and -11 then we can use

```
>> K=place(A,B,[-10 -11])
K =
    26.0000    72.0000
```

This means that  $K = [26 \ 72]$







**Example 4.7**

$$A = \begin{bmatrix} -20 & -225 & -1250 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = [1 \ 0 \ 0]^T, \quad C = [0 \ 0 \ 1], \quad D = \mathbf{0}, \quad \text{Desired}$$

closed loop location:  $P = [-50 \ -10 + 10j \ -10 - 10j]$ ,  $(x(0) = [1 \ 1 \ 0])$

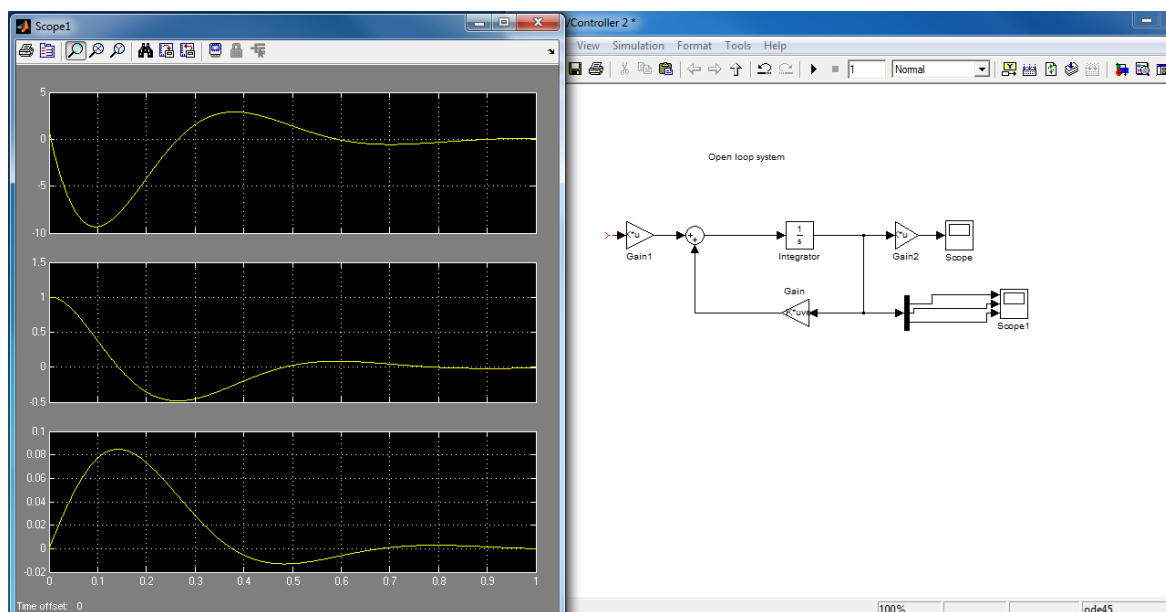
```
>> A = [-20 -225 -1250; 1 0 0; 0 1 0]; B = [1 0 0]'; C = [0 0 1];
D = 0;
```

```
>> eig(A)
```

```
ans =
```

```
-5.0000 +10.0000i
-5.0000 -10.0000i
-10.0000
```

Stable but not very fast!!



To make the system faster we use a state feedback law but we need first to check controllability:

```
>> rank(ctrb(A,B))
```

ans =

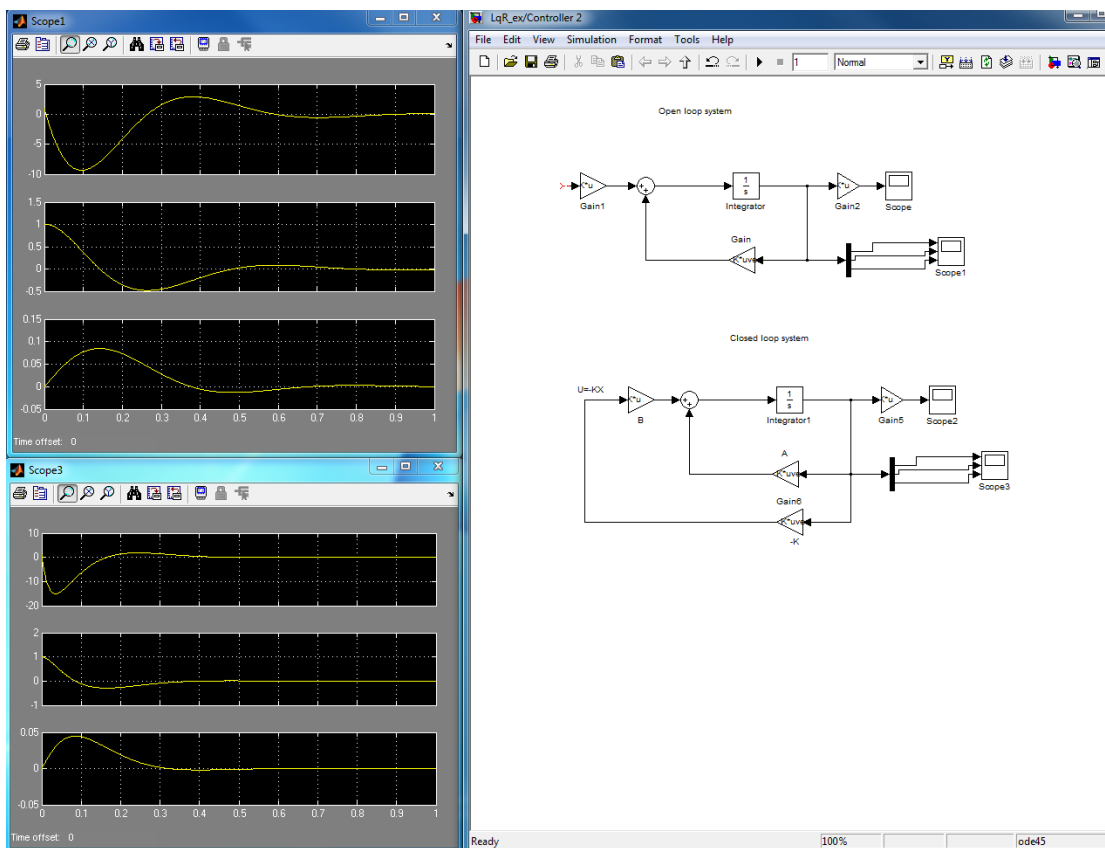
3

```
>> K=place(A,B,[-50 -10+10j -10-10j])
```

K =

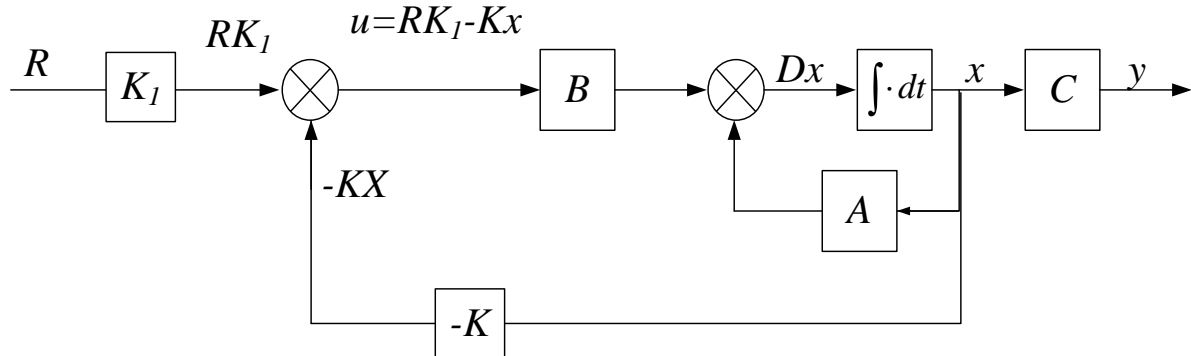
1.0e+003 \*

0.0500 0.9750 8.7500

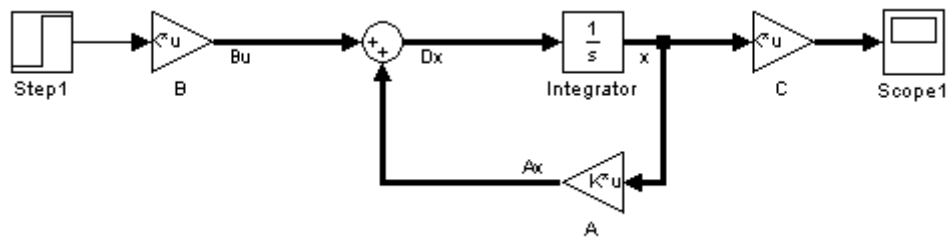


**Example 4.8 with a non zero input:**

The overall block diagram is:

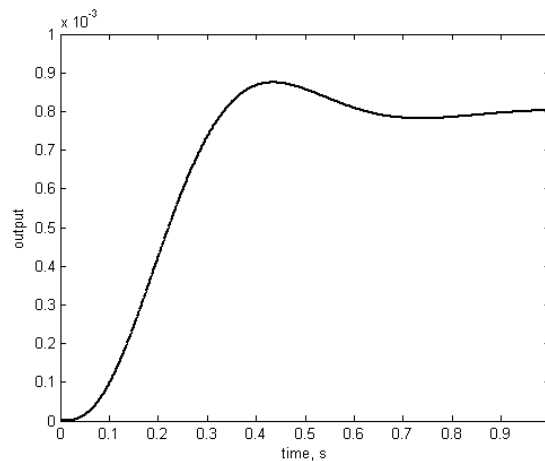


$$A = \begin{bmatrix} -20 & -225 & -1250 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = [1 \ 0 \ 0]^T, C = [0 \ 0 \ 1], D = 0, (x(0) = 0_3).$$



Find the open loop step response then design a pole placement controller with poles located at -10, -20 and -30.

OL step response ( $u=1$ ):



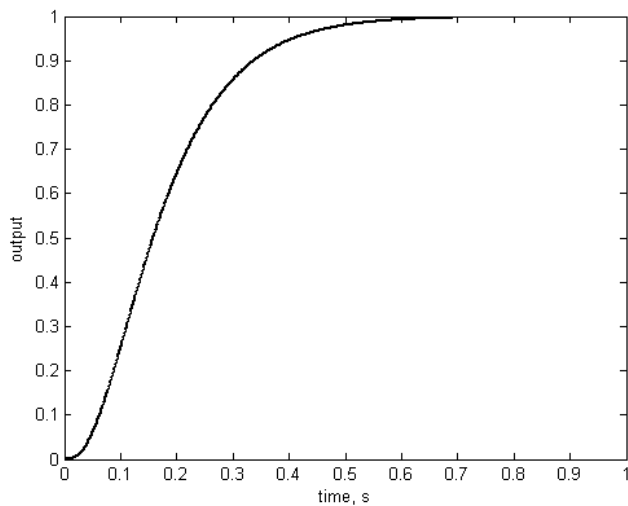
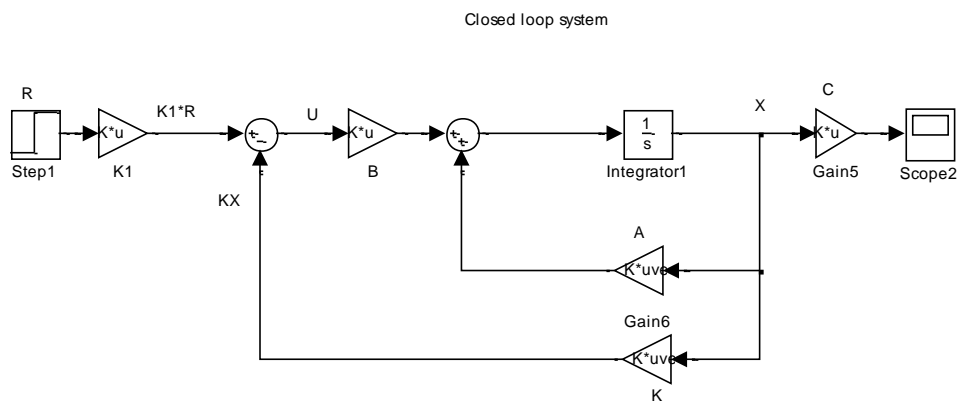
```
>> A=[-20 -225 -1250;1 0 0;0 1 0];B=[1;0;0];
>> rank(ctrb(A,B))
```

```
ans =
     3
>> K=place(A,B,[-10 -20 -30])
```

```
K =
```

```
1.0e+003 *
    0.0400    0.8750    4.7500
```

The value of  $K_1$ , i.e. the value that we need to get a steady state of 1 is found after trial and error:  $K_1=6000$  then:



### 3. Linear Quadratic Regulator (LQR)

What is the correct location of the poles, i.e. a compromise between speed and energy (or control effort) that we use?

**Task:** Design a controller (called LQR)  $u = -Kx$  for  $\dot{x} = Ax + Bu$  such as it is going to minimise:  $J = \int_0^{\infty} (x^T Q x + u^T R u) dt$ , where  $Q$  and  $R$  are positive definite matrices.  $Q$ : Importance of the error,  $R$ : Importance of the energy that we use.

Using Matlab:  $[K, P, E] = \text{lqr}(A, B, Q, R)$ .  $E$  matrix gives the eigenvalues of  $A - BK$  and  $K$  is the state feedback control gain. You can then simulate the LQR as a normal state feedback control with a gain of  $K$ .

#### Example 4.9:

$A = \begin{bmatrix} 0 & 1 \\ 0 & -1 \end{bmatrix}$ ,  $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$  Find  $K$ , the eigenvalues of  $A - BK$ , and the response of the system for  $R=1$  and  $Q=\text{eye}(2)$  and  $Q=10*\text{eye}(2)$  ( $x(0)=[1 \ 1]$ ).

```
>> A=[0 1;0 -1]; B=[0;1]; R=1; Q1=eye(2);
Q2=10*eye(2);
```

```
>> rank(ctrb(A,B))
```

```
ans =
```

```
2
```

```
>> [K1,P1,E1]=lqr(A,B,Q1,R)
```

```
K1 =
```

```
1.0000 1.0000
```

P1 =

```

    2.0000    1.0000
    1.0000    1.0000

```

E1 =

```

   -1.0000
   -1.0000

```

```
>> [K2, P2, E2] = lqr(A, B, Q2, R)
```

K2 =

```

    3.1623    3.1623

```

P2 =

```

   13.1623    3.1623
    3.1623    3.1623

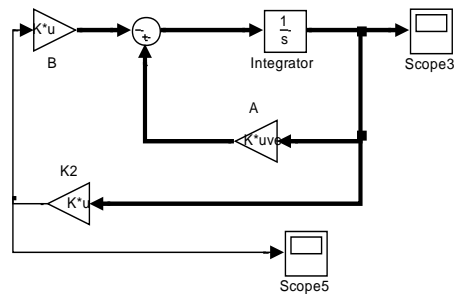
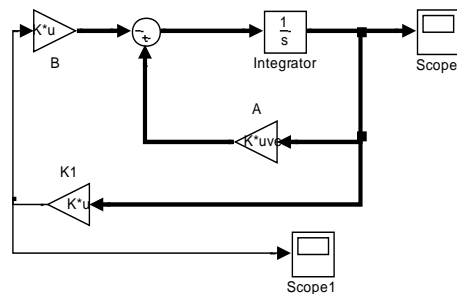
```

E2 =

```

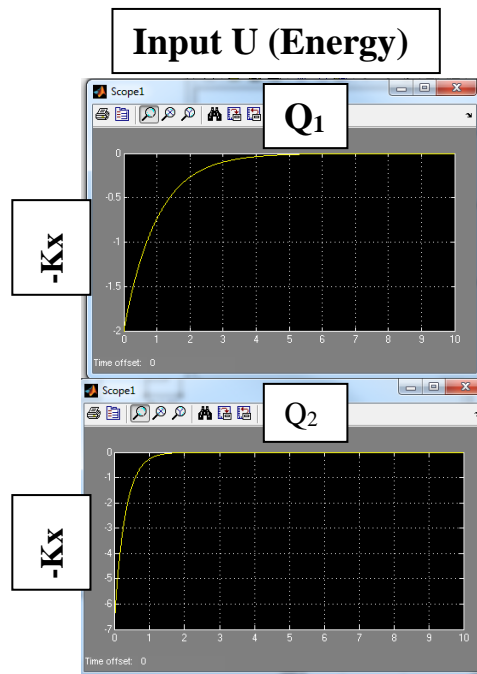
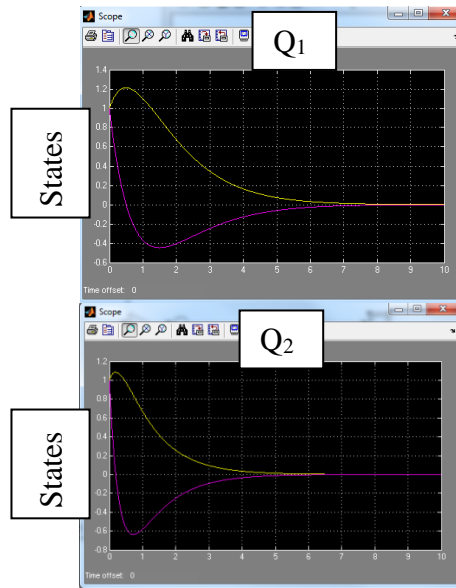
   -1.0000
   -3.1623

```



**State Response**

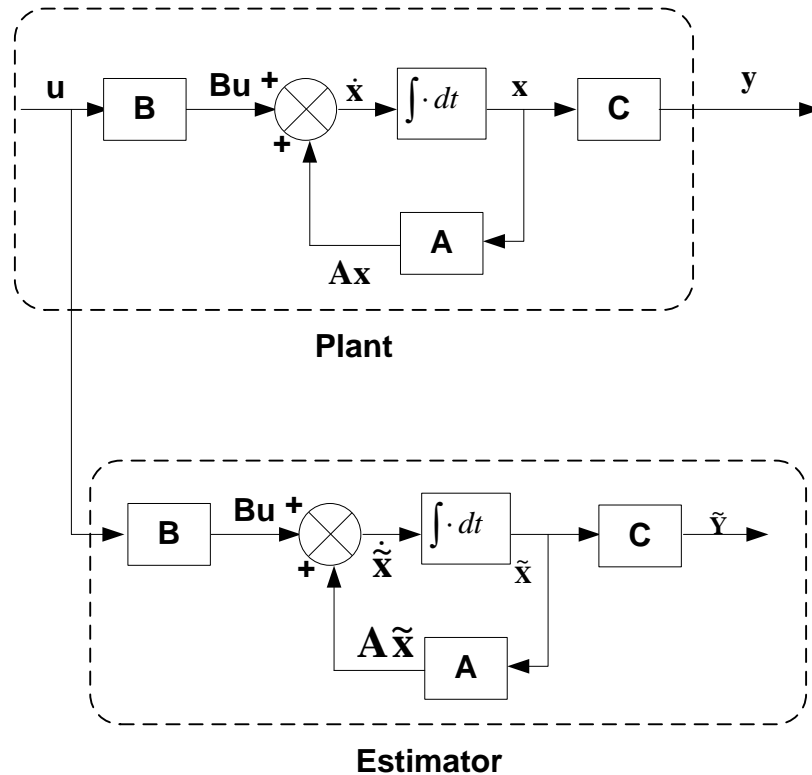




$Q_2$  is faster but more energy is needed. So we can see that  $U$  has increased when we increased  $Q$  to give faster response. ■

## 4. Estimating Techniques

**Open loop estimator:**



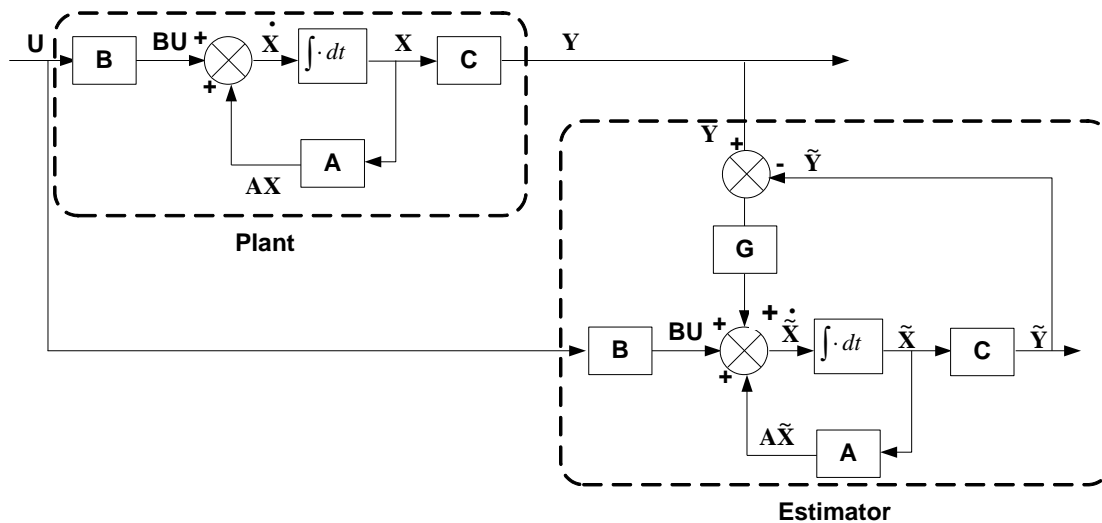
The error between the estimated and real states is

$$e(t) = x(t) - \tilde{x}(t) \Rightarrow \dot{e}(t) = \dot{x}(t) - \dot{\tilde{x}}(t) \Leftrightarrow$$

$$Ax(t) + Bu(t) - A(t)\tilde{x}(t) - Bu(t) \Rightarrow \dot{e}(t) = Ae(t)$$

**Closed loop estimator:**

If we use the error between the actual and the estimated output  $y - \tilde{y}$  and feed this signal back into the estimator we form a **closed loop estimator**.



$$\dot{e}(t) = A(x(t) - \tilde{x}(t)) - GC(x(t) - \tilde{x}(t)) = Ae(t) - GCe(t) = (A - GC)e$$

Hence by appropriately choosing  $G$  we can force the error to converge to zero very fast. If  $G$  is designed such that  $(A - GC)$  is stable, then  $\tilde{x}(t)$  will converge asymptotically towards  $x(t)$ .

### **But the system must be observable!!**

State estimators can be designed **if and only if** the observability condition is satisfied.

### **How to check the performance of the estimation process??**

We can use different methods to measure the estimation error. Here we will use 2 quantities to assess the estimation performance of the estimator: the *absolute estimation error* and the *Integral Absolute of the estimation Error* (IAE).

The absolute estimation error for each state is:  $|e_i(t)| = |x_i(t) - \tilde{x}_i(t)|$  for  $i=1,2,\dots,n$ .

The Integral Absolute of the estimation Error (IAE) is:  $IAE = \sum_{i=1}^n \int |e_i(t)| dt$  so for a 2<sup>nd</sup> order system:

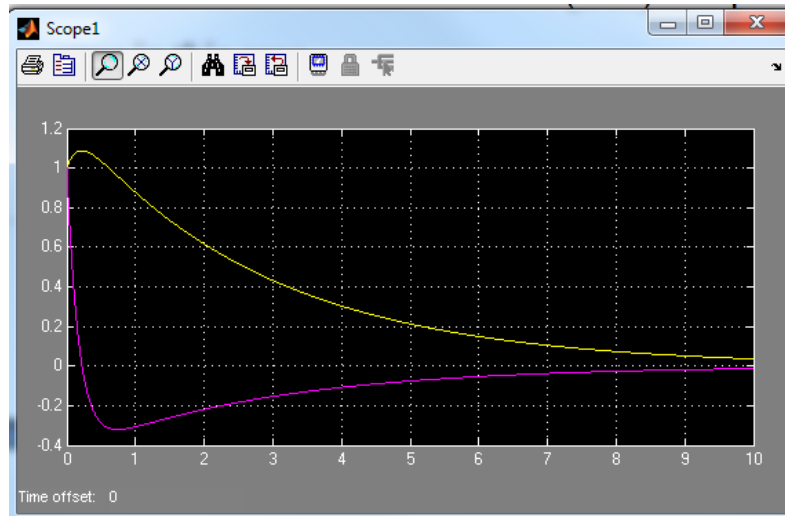
$$IAE = \sum_{i=1}^2 \int |e_i(t)| dt = \int |e_1(t)| dt + \int |e_2(t)| dt = \int |x_1(t) - \tilde{x}_1(t)| dt + \int |x_2(t) - \tilde{x}_2(t)| dt$$

### Example 4.10:

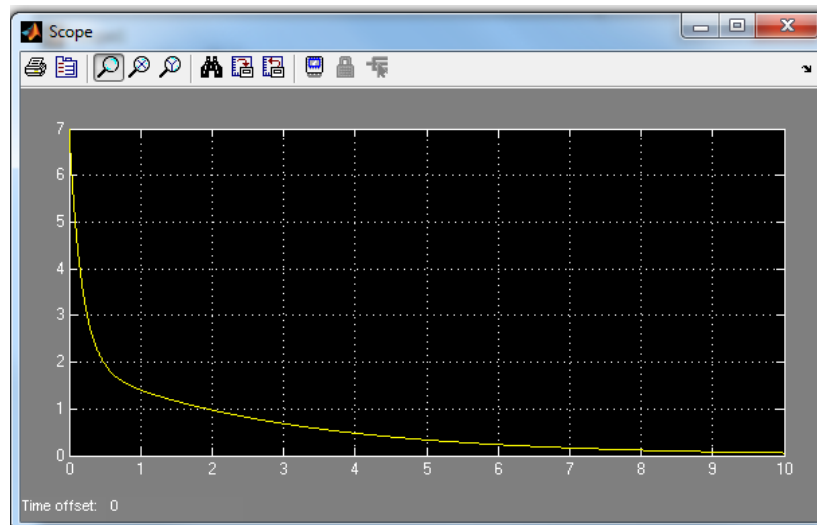
Use an open loop estimator to estimate the states of the state space system given by  $\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -2 & -6 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ ,  $\mathbf{C} = [3 \quad 4]$ ,  $\mathbf{x}_0 = [1 \quad 1]^T$  assuming that the initial condition of the estimated state vector is  $\tilde{\mathbf{x}}_0 = [0.95 \quad 0.95]^T$ . Find the Integral Absolute of the estimation Error (IAE) and plot the absolute state estimation error (Assume  $u(t)=0$ ). Design a closed loop estimator for the system assuming that the estimator poles are located at -12, -15. Find the Integral Absolute of the estimation Error (IAE) and plot the absolute state estimation error in this case then design a state feedback control with poles at -4,-6.

**Original system:**

**State Response**

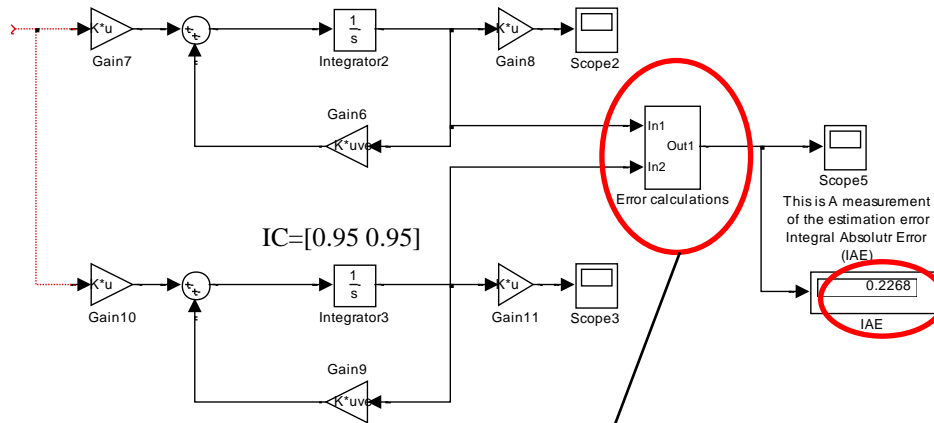


### Output Response

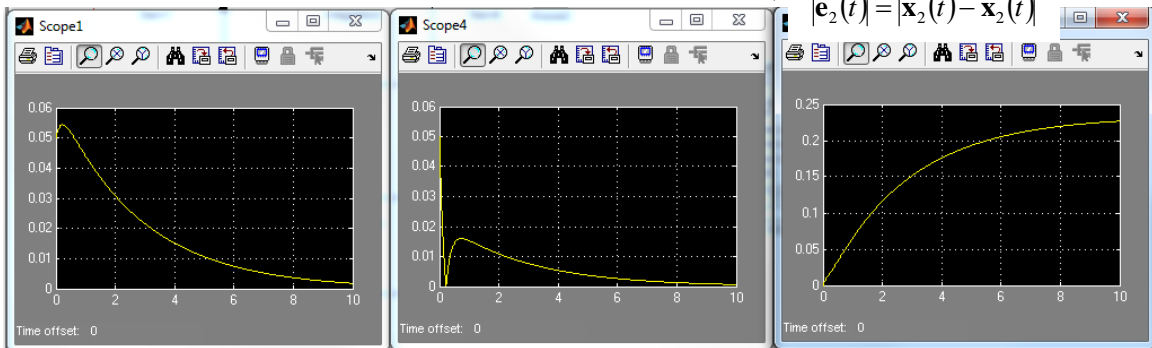
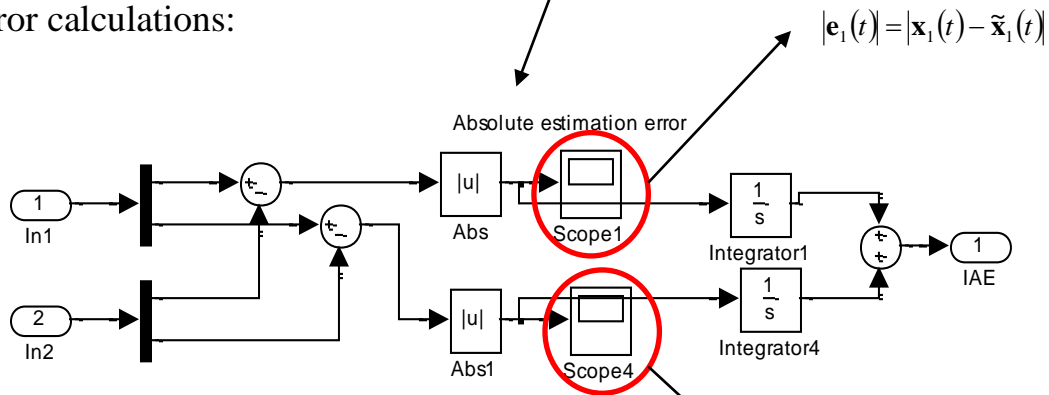


**Open loop estimator:**

$$IC=[1 \ 1]$$



Error calculations:



$$|e_1(t)| = |x_1(t) - \tilde{x}_1(t)|$$

(Scope 1)

$$|e_2(t)| = |x_2(t) - \tilde{x}_2(t)|$$

(Scope 4)

$$IAE = \underline{\underline{0.2268}}$$

**Closed loop estimator:**

**First we need to check observability!!**

```
>> A=[0 1;-2 -6];B=[1;3];C=[3 4];
```

```
>> rank(observ(A,C))
```

```
ans =
```

```
2
```

Hence the system is observable. Then we can design an estimator for that system. To locate the poles of the estimator at  $-12$ ,  $-15$  ( $\lambda_1$  and  $\lambda_2$ ) we need to find  $\mathbf{G}$ .

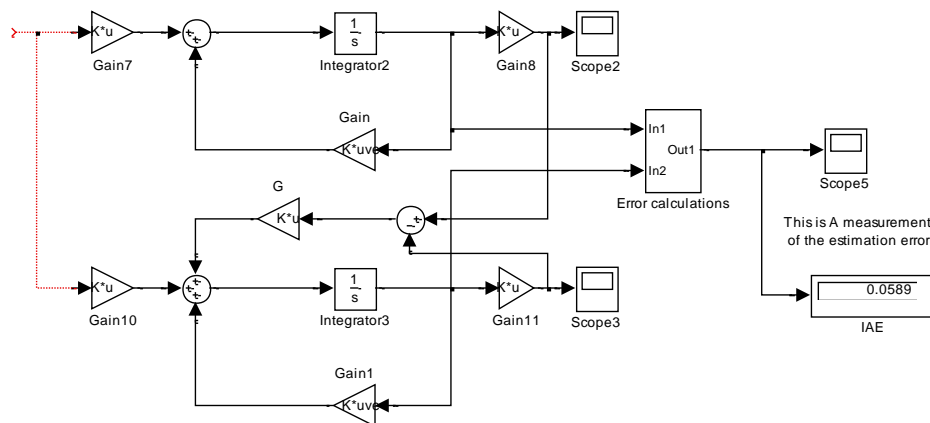
In Matlab we can use the command  $\mathbf{G}=\text{place}(\mathbf{A}',\mathbf{C}',[\lambda_1 \lambda_2])'$  to find  $\mathbf{G}$  where ( $\lambda_1$  and  $\lambda_2$ ) are the eigenvalues of  $\mathbf{A}-\mathbf{GC}$

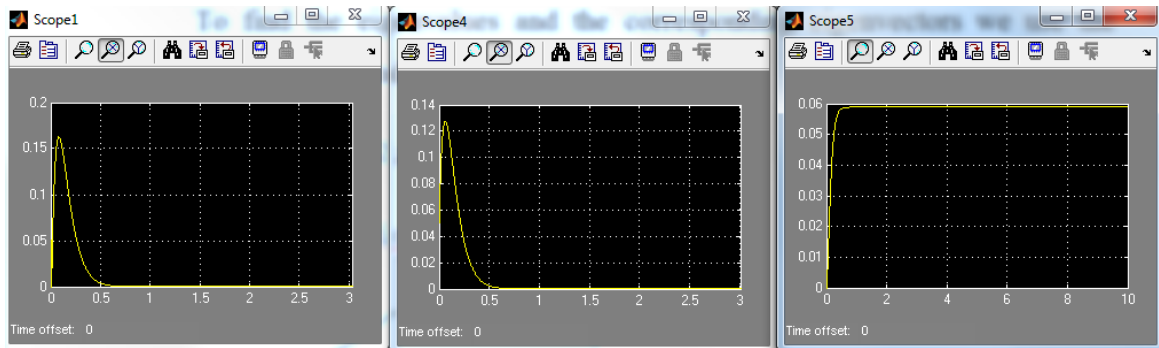
```
>> G=place(A',C',[-12 -15])'
```

```
G =
```

```
20.9355
```

```
-10.4516
```





$$|e_1(t)| = |x_1(t) - \hat{x}_1(t)|$$

$$|e_2(t)| = |x_2(t) - \hat{x}_2(t)|$$

$$\text{IAE} = \underline{\underline{0.0589}}$$

By designing a closed loop estimator we can have better error dynamics (estimation error converges to zero very fast) and hence less IAE compared to the open loop estimator case. Then by properly choosing  $G$  the closed loop estimator provides better estimation performance than the open loop estimator.

- Design a state feedback controller with poles located at  $-4, -6$  using the open loop and closed loop estimators.

```
>> eig(A)
```

```
ans =
```

```
-0.3542
```

```
-5.6458
```

**We need to check if the system is Controllable??**

```
>> rank(ctrb(A, B))
```

```
ans =
```

```
2
```



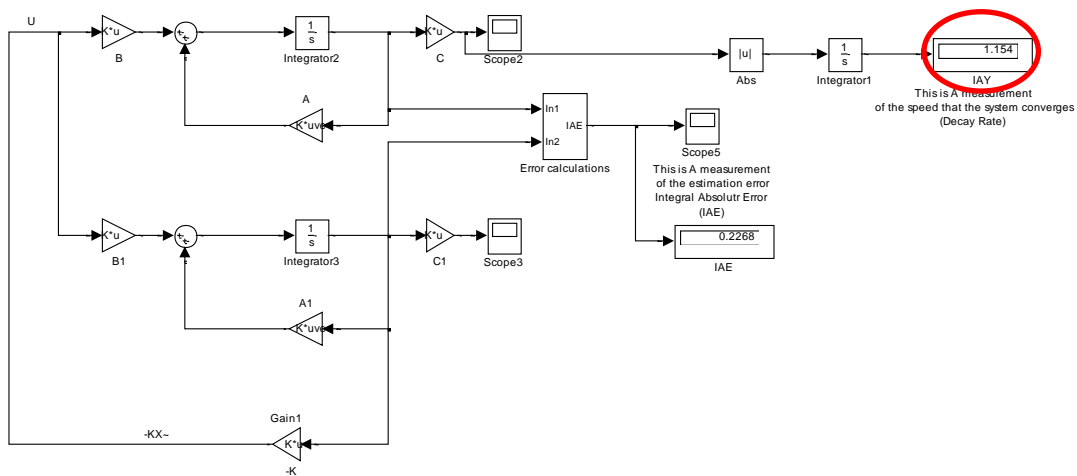
Then the system is controllable. To find the  $\mathbf{K}$  matrix for the state feedback control we use the command *place*:

```
>> K=place(A,B,[-4 -6])
```

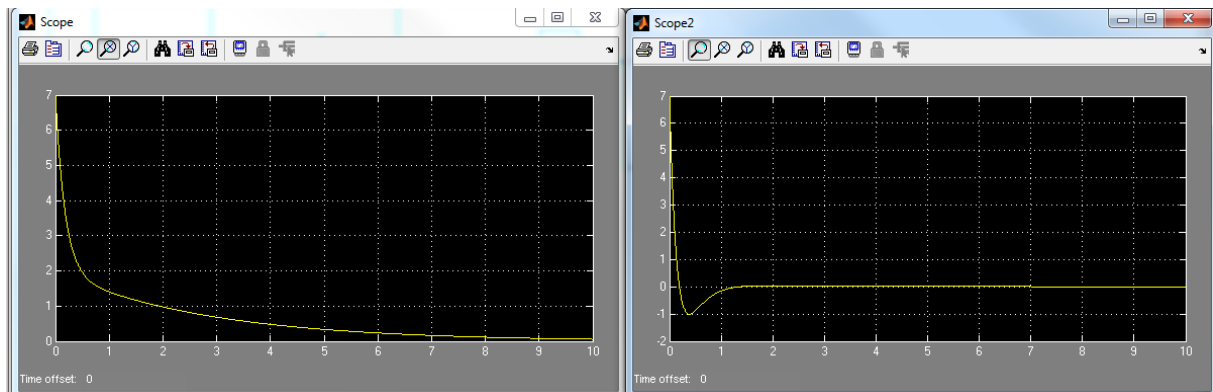
$\mathbf{K} =$

2.5517      0.4828

**System+OL estimator+controller**



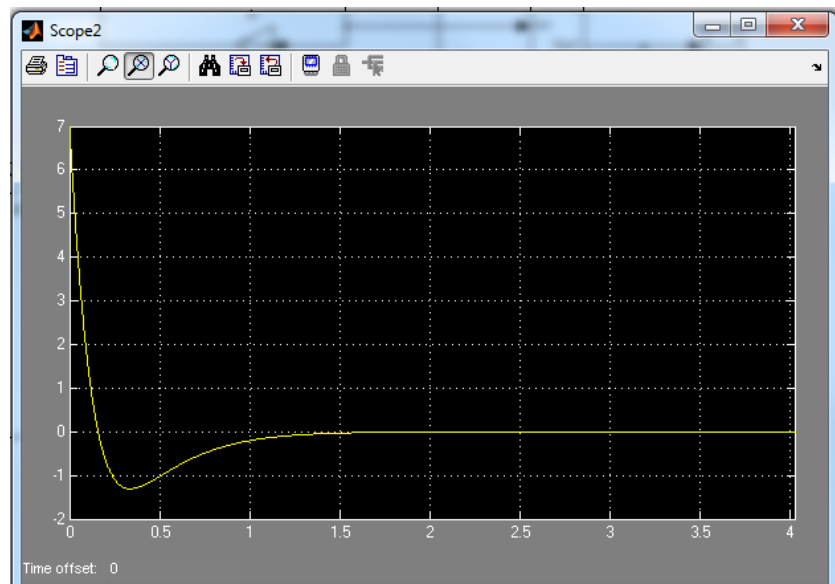
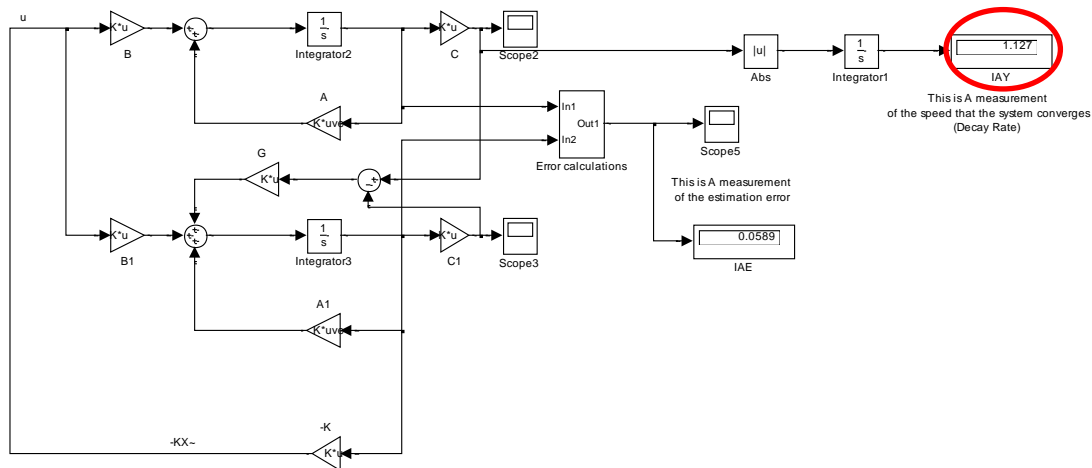
**Output response**



Original system

System+OL estimator+Controller

## System+CL estimator+controller



Output response  
System+CL estimator+Controller

To check which system shows better output response we use IAY (Integral absolute of the output) that gives a measure of the speed of convergence (decay rate). Small values show fast convergence and hence better performance.

**Exercise:**

Check the stability of the state space system: 
$$\left. \begin{aligned} \dot{x}(t) &= \begin{bmatrix} 1 & 3 \\ 2 & -4 \end{bmatrix} x(t) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u \\ y(t) &= [1 \quad 1] x(t) \end{aligned} \right\} .$$

- Simulate the state space system and plot the output response assuming  $\underline{u}(t)=0$  and  $x(0)=[1 \ 0]^T$ .
- Design a closed loop estimator with poles located at -50,-60.
- Simulate the system with the closed loop estimator assuming  $\tilde{x}_0 = [0.95 \ 0]^T$ . Find the Integral Absolute of the estimation Error (IAE) and plot the absolute state estimation error.
- If the system is unstable create a state feedback controller that will stabilise the system. Suggest a proper location for the closed loop poles of the control system.

$$\Rightarrow G = \begin{bmatrix} 699 \\ -592 \end{bmatrix}$$

$$\Rightarrow K = [5.33 \quad 2.66]$$

Now simulate the overall system with the state feedback controller and the closed loop estimator then plot the output response before and after compensation. Find the Integral Absolute of the Output IAY for each case.

## State Space Analysis and Controller Design

### EEE8013

### Tutorial Exercise IV

1. A system is described by  $\mathbf{A} = \begin{bmatrix} 2 & 2 \\ 2 & 5 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ ,  $\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

- Is it stable?
- Find the system's response ( $\mathbf{x}(0)=1$ ,  $\mathbf{x}'(0)=0$ ) and hence crosscheck your answer.
- Stabilise the system using a pole placement controller. Place the poles of the controller at -5,-6.
- Simulate the closed loop system and sketch the output response.

2. Repeat exercise 1 for  $\mathbf{A} = \begin{bmatrix} -2 & 2 \\ 2 & 5 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ ,  $\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ .

3. A system is described by  $\mathbf{A} = \begin{bmatrix} -2 & 2 \\ 2 & -5 \end{bmatrix}$ ,  $\mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ ,  $\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

- Is it stable?
- Find the system's response ( $\mathbf{x}(0)=1$ ,  $\mathbf{x}'(0)=0$ ) and hence crosscheck your answer.
- Use a state feedback controller that will locate the closed loop poles at -10,-12. Plot the output response.
- Use a state feedback controller that will locate the closed loop poles at -30,-50. Plot the output response.
- Plot the signal  $-\mathbf{Kx}$  in the previous 2 cases. What do you notice?
- Design an LQR for the system with the following specifications

$$\mathbf{Q} = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}, R = 1. \text{ Find the new eigenvalues and plot the output response in}$$

this case and  $-\mathbf{Kx}$  signal

- Design another LQR for the system with the following specifications

$$\mathbf{Q} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}, R = 1. \text{ Find the new eigenvalues and plot the output response}$$

in this case and  $-\mathbf{K}\mathbf{x}$  signal

4. Repeat question 3 for  $\mathbf{A} = \begin{bmatrix} -2 & 2 \\ 0 & -5 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

5. A system is described by  $\mathbf{A} = \begin{bmatrix} -2 & 2 \\ 2 & -5 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{C} = [1 \quad 1].$

- Find the system's response ( $\mathbf{x}(0)=1, \mathbf{x}'(0)=0$ ) and comment on the output signal. (Assume  $u(t)=0$ )
- Use an open loop estimator to estimate the 2 states when the initial condition of the estimated state vector is  $\tilde{\mathbf{x}}_0 = [0.99 \quad 0]^T$ . Plot the absolute estimation error for each state and the *Integral Absolute of the estimation Error (IAE)*.
- Design a state feedback controller using the above estimator to locate the poles of the closed loop system at  $[-10 \quad -60]$  then create another controller to locate the poles at  $[-0.5 \quad -3]$ . Simulate the overall system in each case and comment on your results.

6. Use a closed loop estimator for  $\mathbf{A} = \begin{bmatrix} 2 & 2 \\ 2 & -5 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{C} = [1 \quad 1]$ . Place the poles of the estimator at  $[-10 \quad -11]$ . Plot the absolute estimation error for each state and the *Integral Absolute of the estimation Error (IAE)*.

7. Create 2 different pole placement control strategies and then use the above estimator:
- One with desired closed loop poles at  $[-5 \quad -6]$
  - One with desired closed loop poles at  $[-15 \quad -16]$ .

Which control law is more appropriate? Comment on your results.

8. Repeat question 6 when  $\mathbf{A} = \begin{bmatrix} 2 & 0 \\ 2 & -5 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{C} = [1 \quad 0]$ .