Automatic Control (1)

By

Associate Prof. / Mohamed Ahmed Ebrahim Mohamed

E-mail: mohamedahmed_en@yahoo.com

mohamed.mohamed@feng.bu.edu.eg

Web site: http://bu.edu.eg/staff/mohamedmohamed033

Lecture (2)

Block Diagram Fundamentals
&
Reduction Techniques
&
Signal-flow Graphs
&
State-Space Representation

Superposition of Multiple Inputs

Step 1: Set all inputs except one equal to zero.

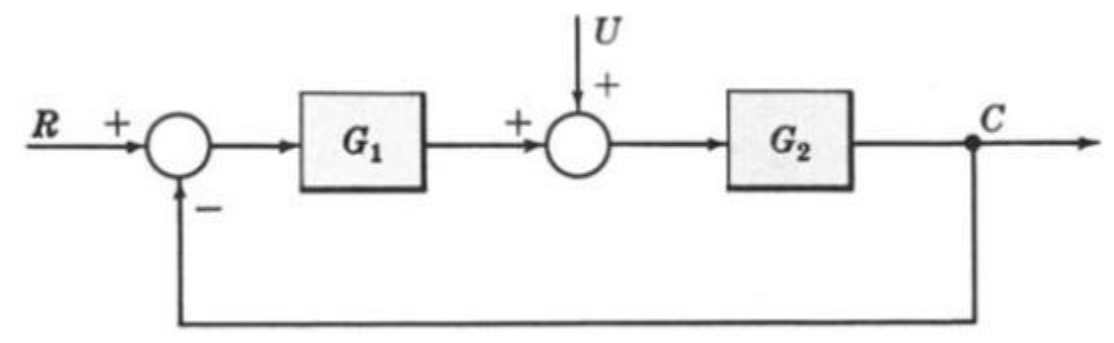
Step 2: Transform the block diagram to canonical form, using the transformations of Section 7.5.

Step 3: Calculate the response due to the chosen input acting alone.

Step 4: Repeat Steps 1 to 3 for each of the remaining inputs.

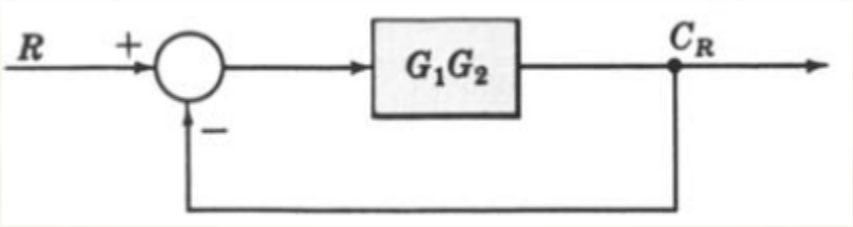
Step 5: Algebraically add all of the responses (outputs) determined in Steps 1 to 4. This sum is the total output of the system with all inputs acting simultaneously.

Example-15: Multiple Input System. Determine the output C due to inputs R and U using the Superposition Method.

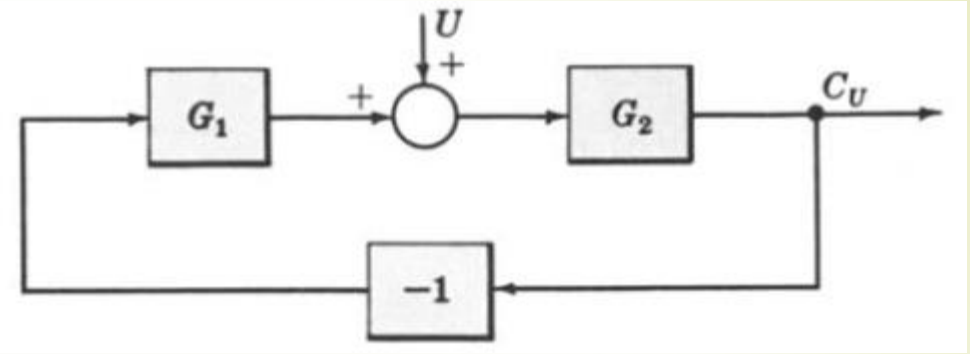


Step 1: Put $U \equiv 0$.

Step 2: The system reduces to



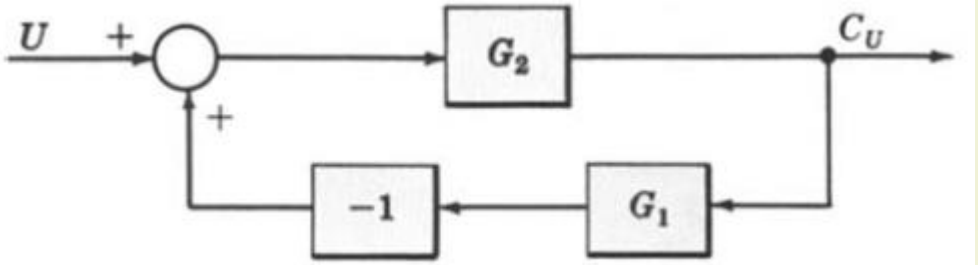
Step 3: the output C_R due to input R is $C_R = [G_1G_2/(1 + G_1G_2)]R$



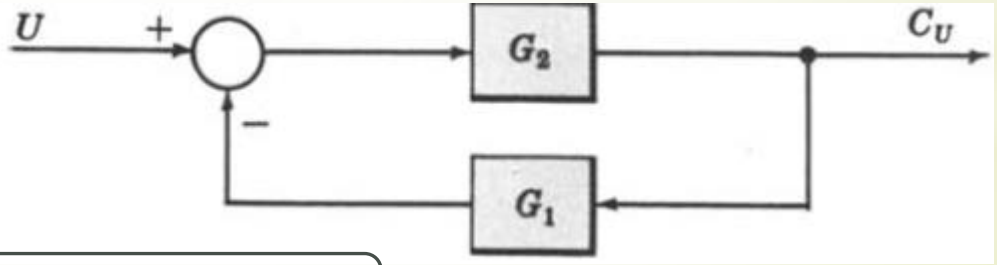
Step 4a: Put $R = 0$.

Step 4b: Put -1 into a block, representing the negative feedback effect:

Rearrange the block diagram:



Let the -1 block be absorbed into the summing point:



Step 4c: the output C_U due to input U is $C_U = [G_2/(1 + G_1G_2)]U$.

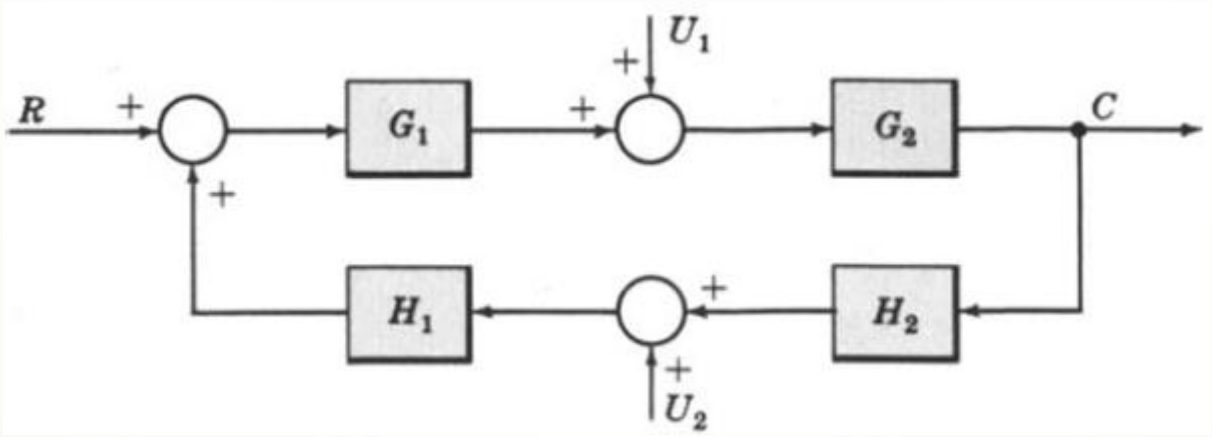
Example-15: Continue

Step 5: The total output is $C = C_R + C_U$

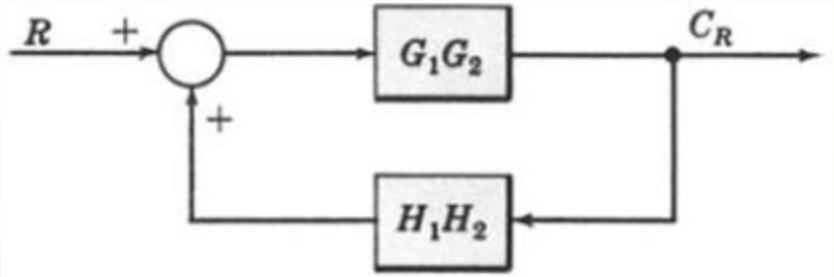
$$= \left[\frac{G_1G_2}{1 + G_1G_2}\right]R + \left[\frac{G_2}{1 + G_1G_2}\right]U$$

$$= \left[\frac{G_2}{1 + G_1G_2}\right][G_1R + U]$$

Example-16: **Multiple-Input System**. Determine the output C due to inputs R, U_1 and U_2 using the Superposition Method.



Let $U_1 = U_2 = 0$.

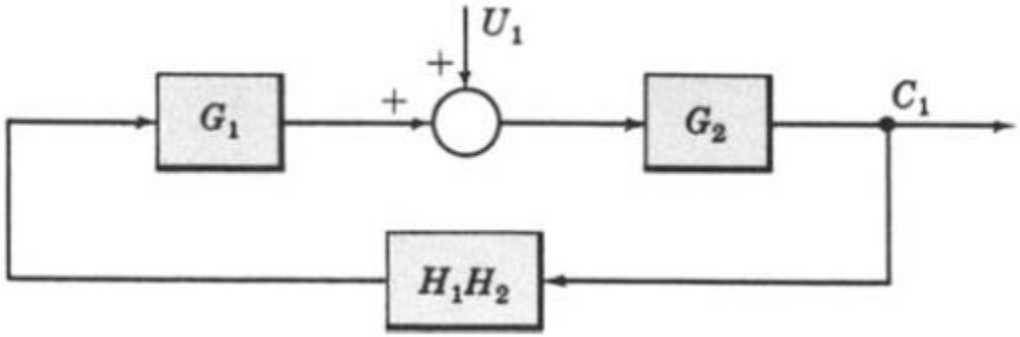


$$C_R = [G_1G_2/(1 - G_1G_2H_1H_2)]R$$

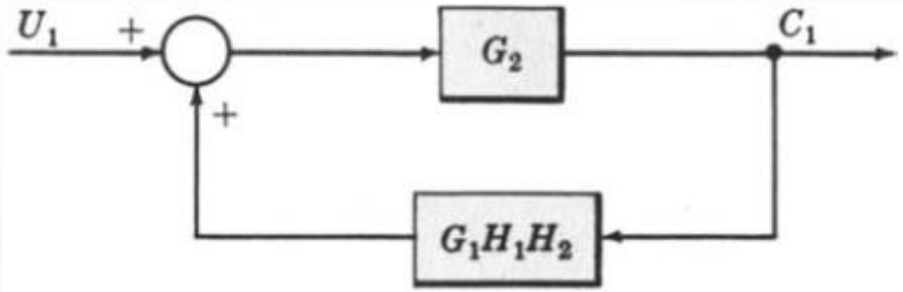
where C_R is the output due to R acting alone.

Example-16: Continue

Now let $R = U_2 = 0$.



Rearranging the blocks, we get

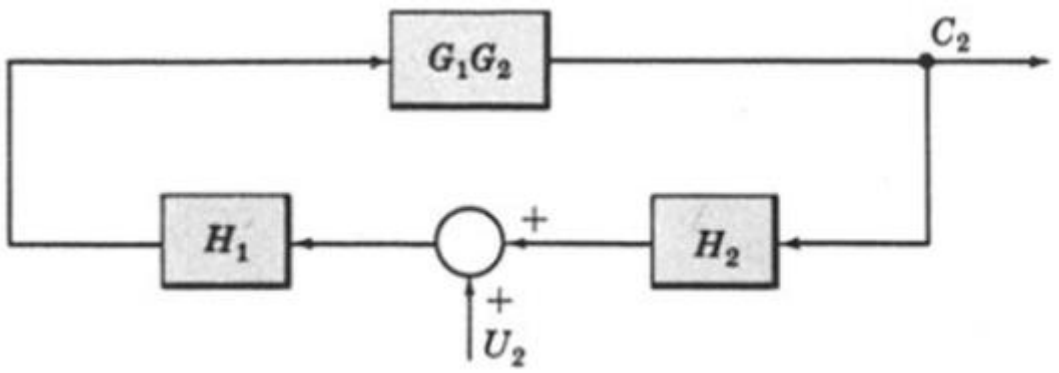


$$C_1 = [G_2/(1 - G_1G_2H_1H_2)]U_1$$

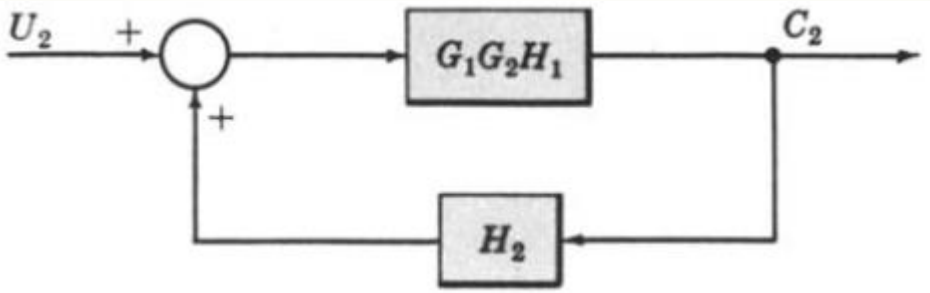
where C_1 is the response due to U_1 acting alone.

Example-16: Continue

Finally, let $R = U_1 = 0$.



Rearranging the blocks, we get



$$C_2 = [G_1G_2H_1/(1 - G_1G_2H_1H_2)]U_2$$

where C_2 is the response due to U_2 acting alone.

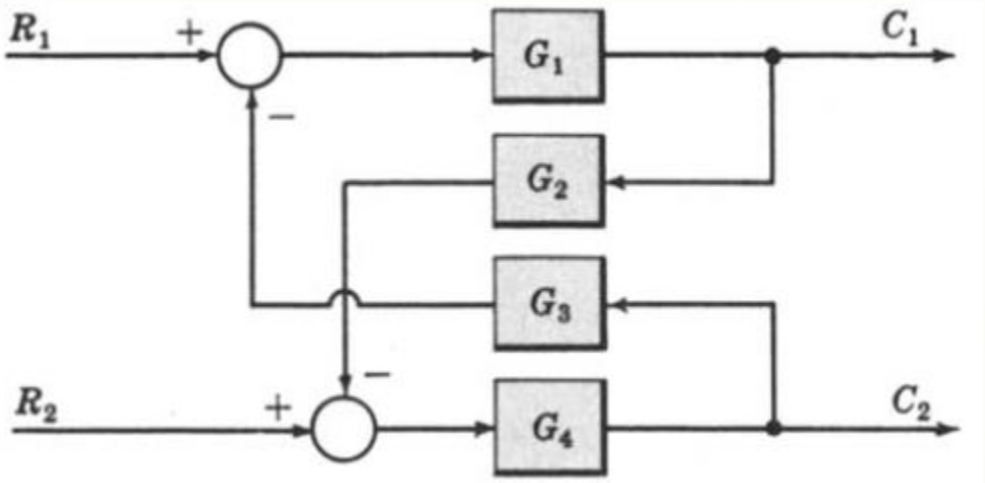
By superposition, the total output is

$$C = C_R + C_1 + C_2 = \frac{G_1G_2R + G_2U_1 + G_1G_2H_1U_2}{1 - G_1G_2H_1H_2}$$

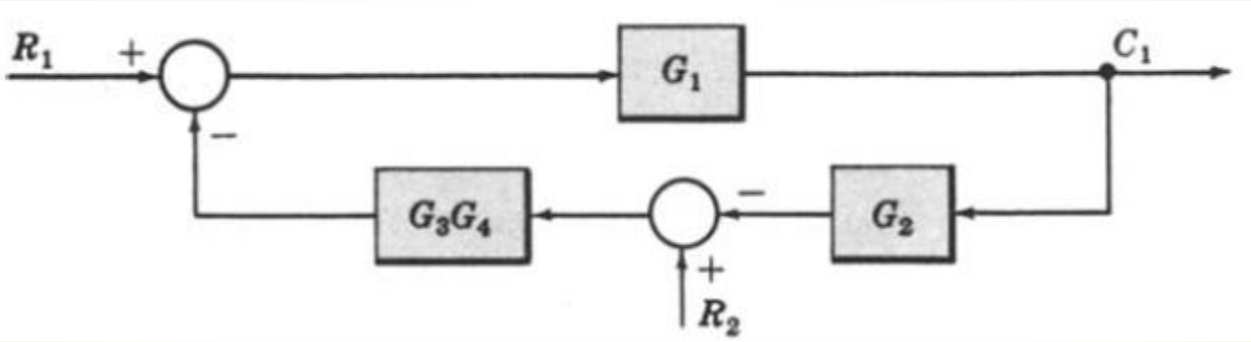
Associate Prof. Dr . Mohamed Ahmed Ebrahim

Example-17: Multi-Input Multi-Output System.

Determine C_1 and C_2 due to R_1 and R_2.

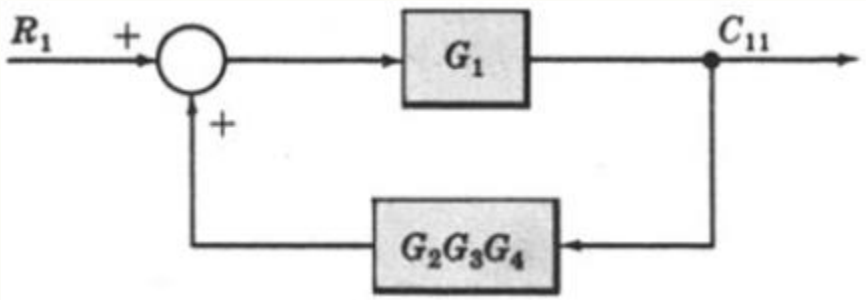


First ignoring the output C_2.



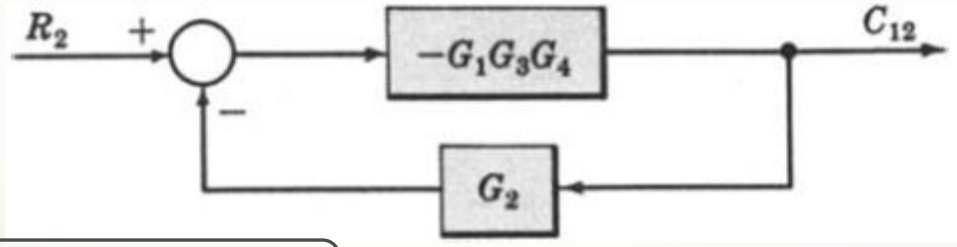
Example-17: Continue

Letting $R_2 = 0$ and combining the summing points,



Hence C_{11}, the output at C_1 due to R_1 alone, is $C_{11} = G_1R_1/(1 - G_1G_2G_3G_4)$.

For $R_1 = 0$,

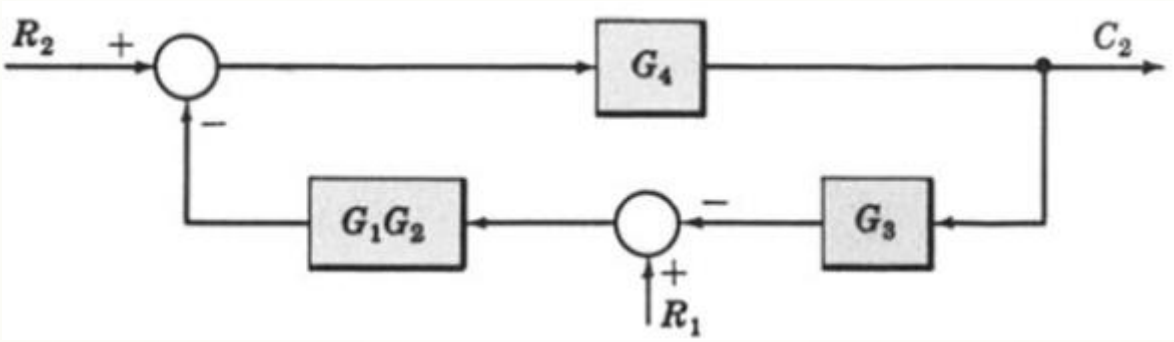


Hence $C_{12} = -G_1G_3G_4R_2/(1 - G_1G_2G_3G_4)$ is the output at C_1 due to R_2 alone.

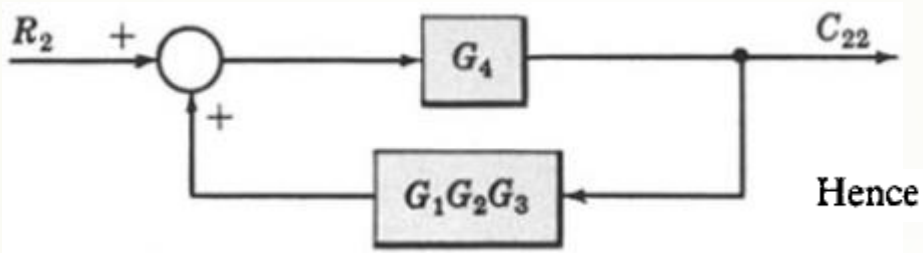
Thus $C_1 = C_{11} + C_{12} = (G_1R_1 - G_1G_3G_4R_2)/(1 - G_1G_2G_3G_4)$

Example-17: Continue

Now we reduce the original block diagram, ignoring output C_1.

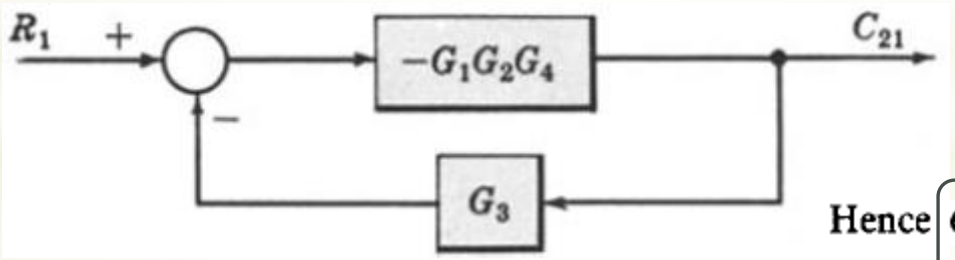


When R1 = 0,



Hence $C_{22} = G_4 R_2/(1 - G_1G_2G_3G_4)$

When R2 = 0,



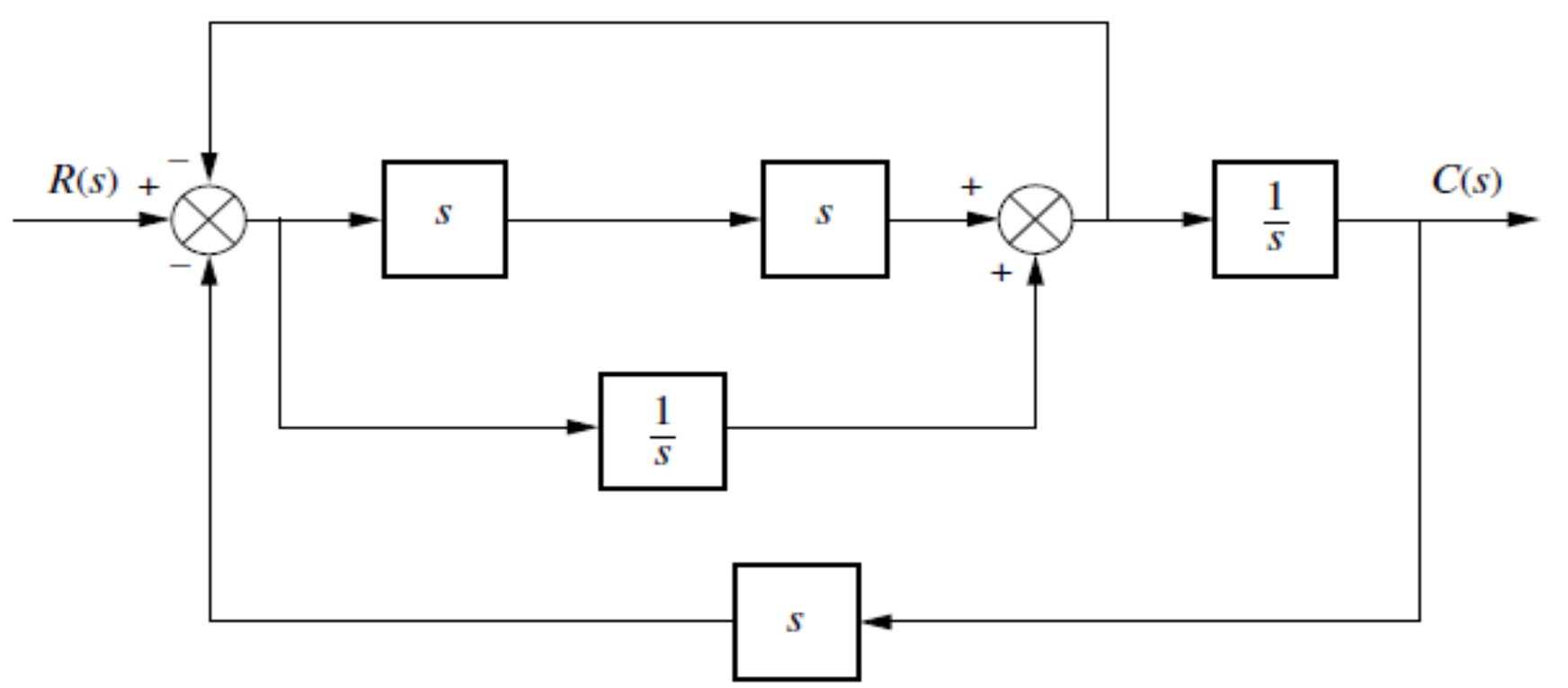
Hence $C_{21} = -G_1G_2G_4 R_1/(1 - G_1G_2G_3G_4)$

Finally, $C_2 = C_{22} + C_{21} = (G_4 R_2 - G_1G_2G_4 R_1)/(1 - G_1G_2G_3G_4)$

Associate Prof. Dr . Mohamed Ahmed Ebrahim

Skill Assessment Exercise:

PROBLEM: Find the equivalent transfer function, $T(s) = C(s)/R(s)$, for the system



Associate Prof. Dr . Mohamed Ahmed Ebrahim

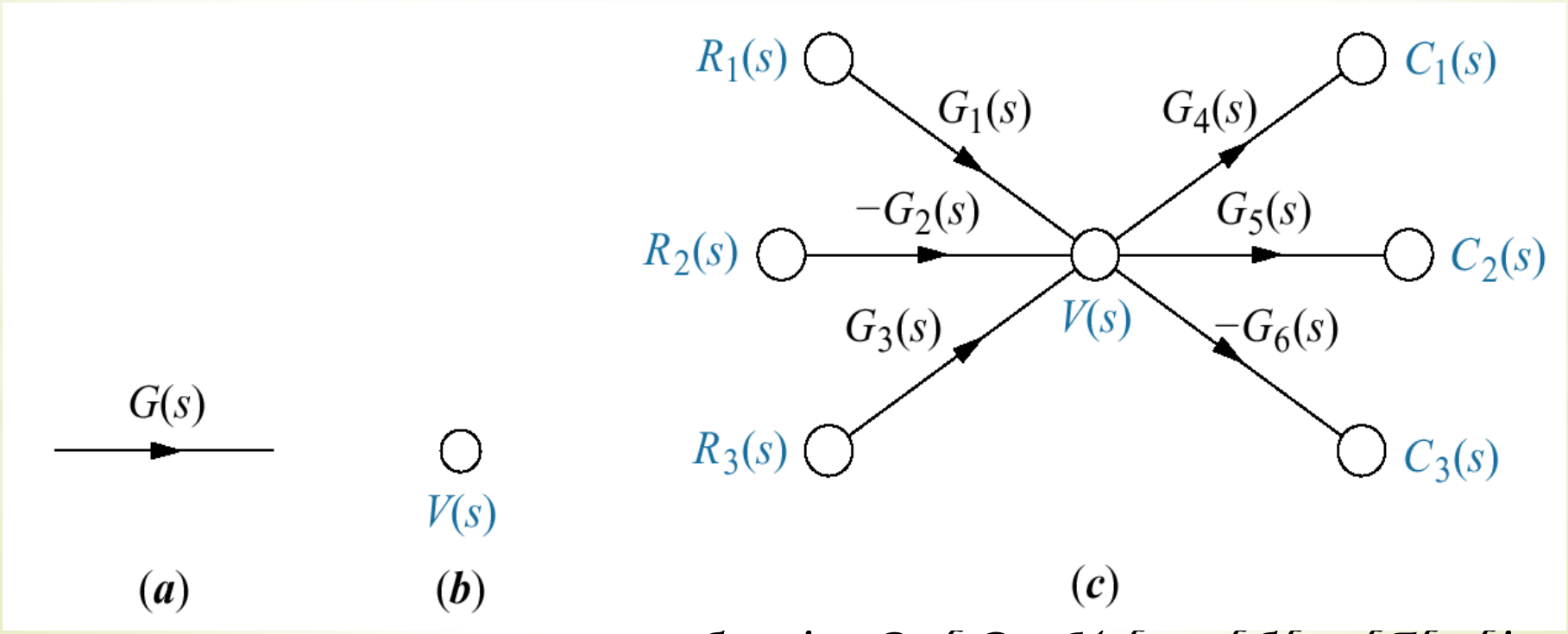
Answer of Skill Assessment Exercise:

ANSWER: $$T(s) = \frac{s^3 + 1}{2s^4 + s^2 + 2s}$$

Associate Prof. Dr . Mohamed Ahmed Ebrahim

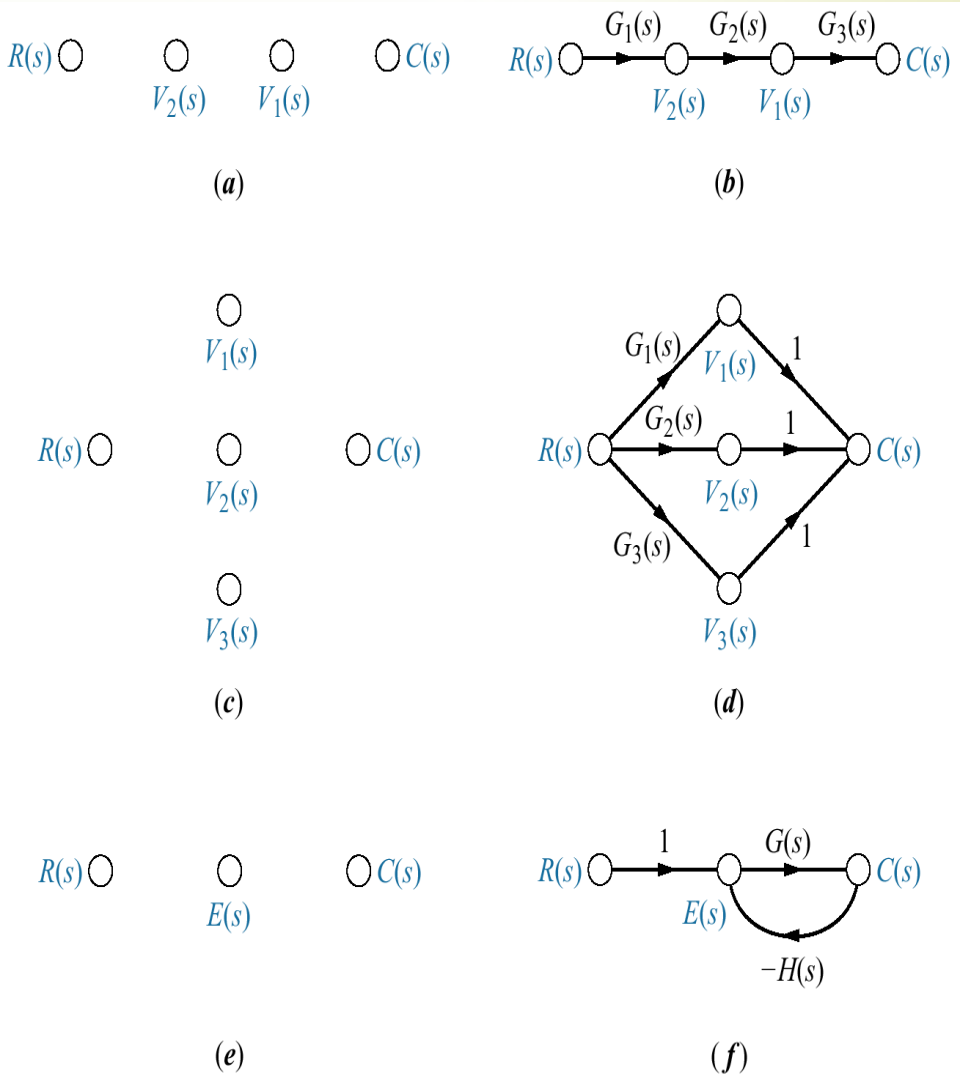
Signal-flow Graph Components

a. system;
b. signal;
c. interconnection of systems and signals



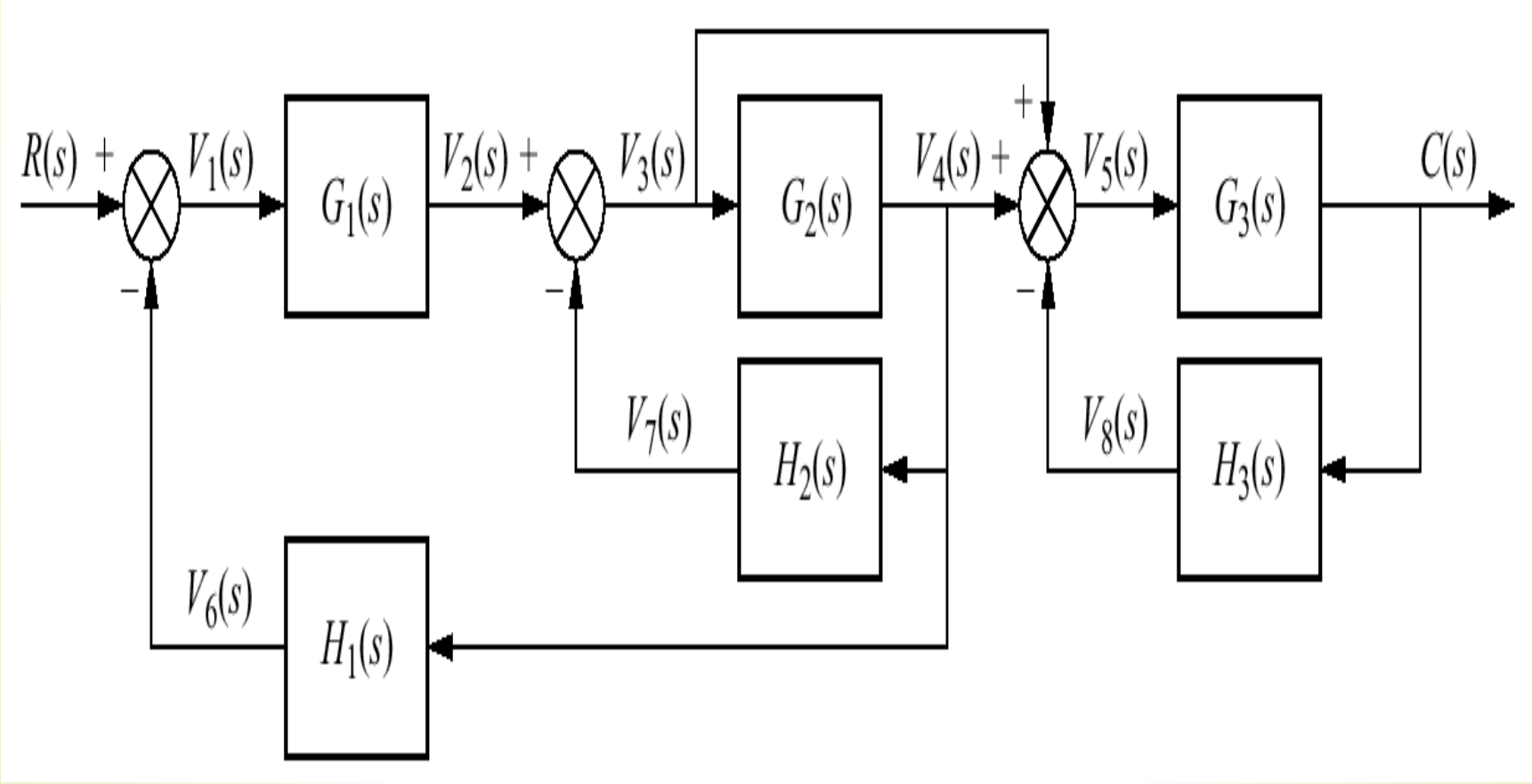
Building Signal-flow Graphs

a. Cascaded system nodes
b. Cascaded system signal-flow graph;
c. Parallel system nodes
d. Parallel system signal-flow graph;
e. Feedback system nodes
f. Feedback system signal-flow graph



Converting a Block Diagram to a Signal-flow Graph

Problem: **Convert the block diagram to a signal-flow graph.**



Associate Prof. Dr . Mohamed Ahmed Ebrahim

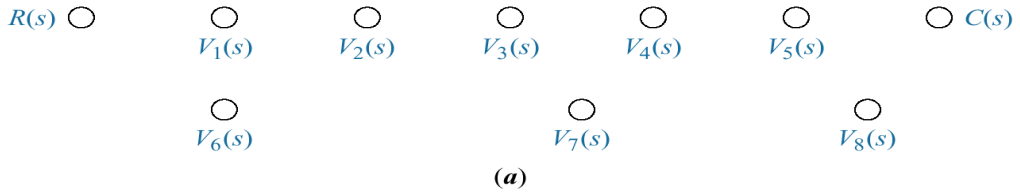
Converting a Block Diagram to a Signal-flow Graph

Signal-flow graph development:

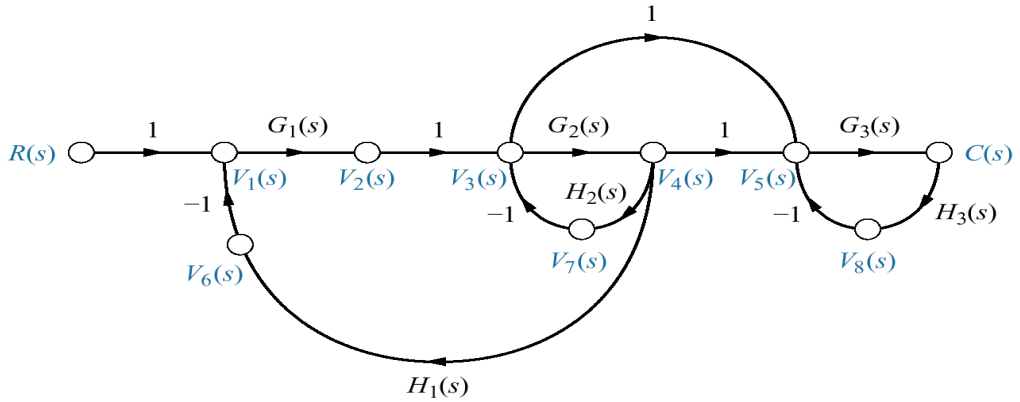
a. signal nodes;

b. signal-flow graph;

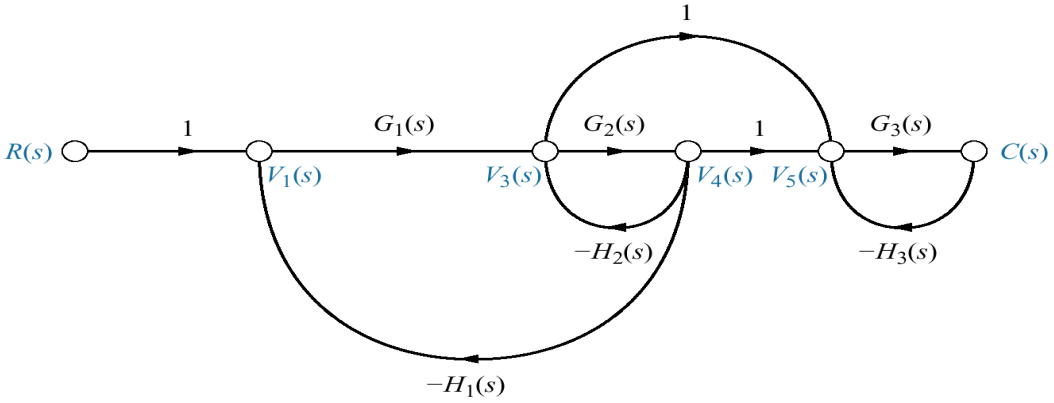
c. simplified signal-flow graph



(a)

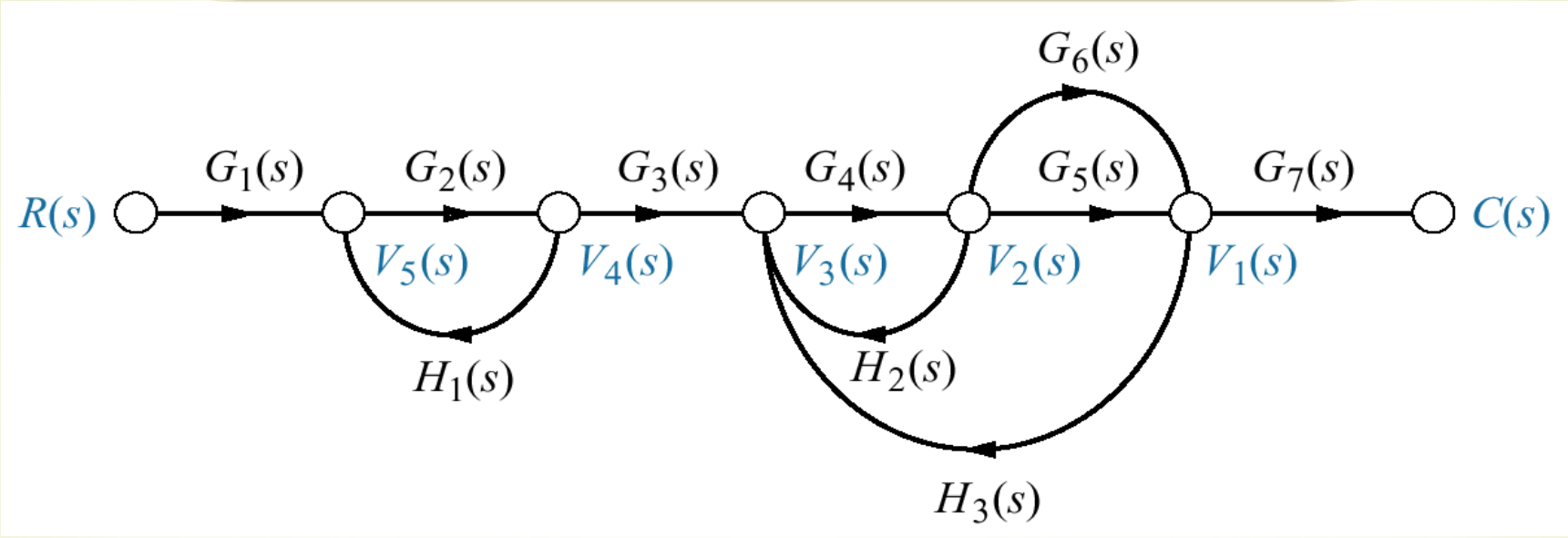


(b)



(c)

Mason's Rule - Definitions

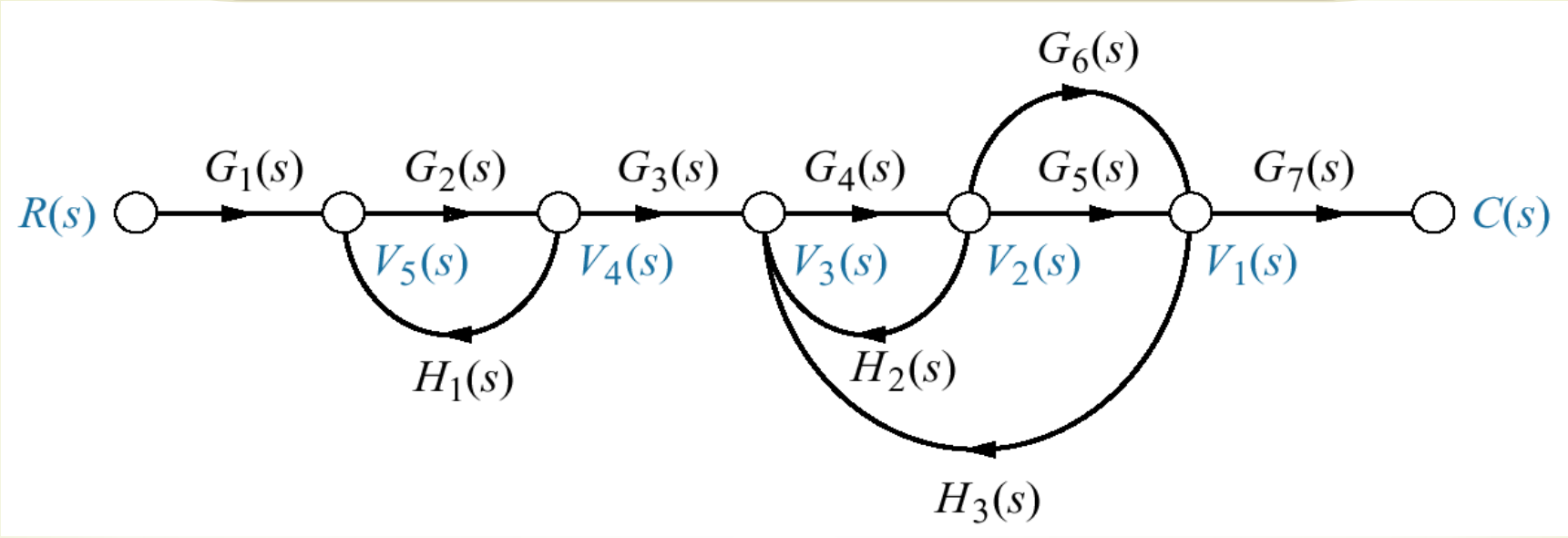


Loop gain "L_k": The product of branch gains found by traversing a path that starts at a node and ends at the same node, following the direction of the signal flow, without passing through any other node more than once. **$G_2(s)H_1(s)$, $G_4(s)H_2(s)$, $G_4(s)G_5(s)H_3(s)$, $G_4(s)G_6(s)H_3(s)$**

Forward-path gain "P_k": The product of gains found by traversing a path from input node to output node in the direction of signal flow. **$G_1(s)G_2(s)G_3(s)G_4(s)G_5(s)G_7(s)$, $G_1(s)G_2(s)G_3(s)G_4(s)G_6(s)G_7(s)$**

Associate Prof. Dr . Mohamed Ahmed Ebrahim

Mason's Rule - Definitions



Nontouching loops: loops that do not have any nodes in common. **$G_2(s)H_1(s)$** does not touch **$G_4(s)H_2(s)$, $G_4(s)G_5(s)H_3(s)$, and $G_4(s)G_6(s)H_3(s)$**

Nontouching-loop gain: The product of loop gains from nontouching loops taken 2, 3,4, or more at a time.

$[G_2(s)H_1(s)][G_4(s)H_2(s)]$, $[G_2(s)H_1(s)][G_4(s)G_5(s)H_3(s)]$, $[G_2(s)H_1(s)][G_4(s)G_6(s)H_3(s)]$

Mason's Rule

The Transfer function. C(s)/ R(s), of a system represented by a signal-flow graph is

$$G(s) = \frac{C(s)}{R(s)} = \frac{\sum_k P_k \Delta_k}{\Delta}$$

Where

K = number of forward paths

P_k = the kth forward-path gain

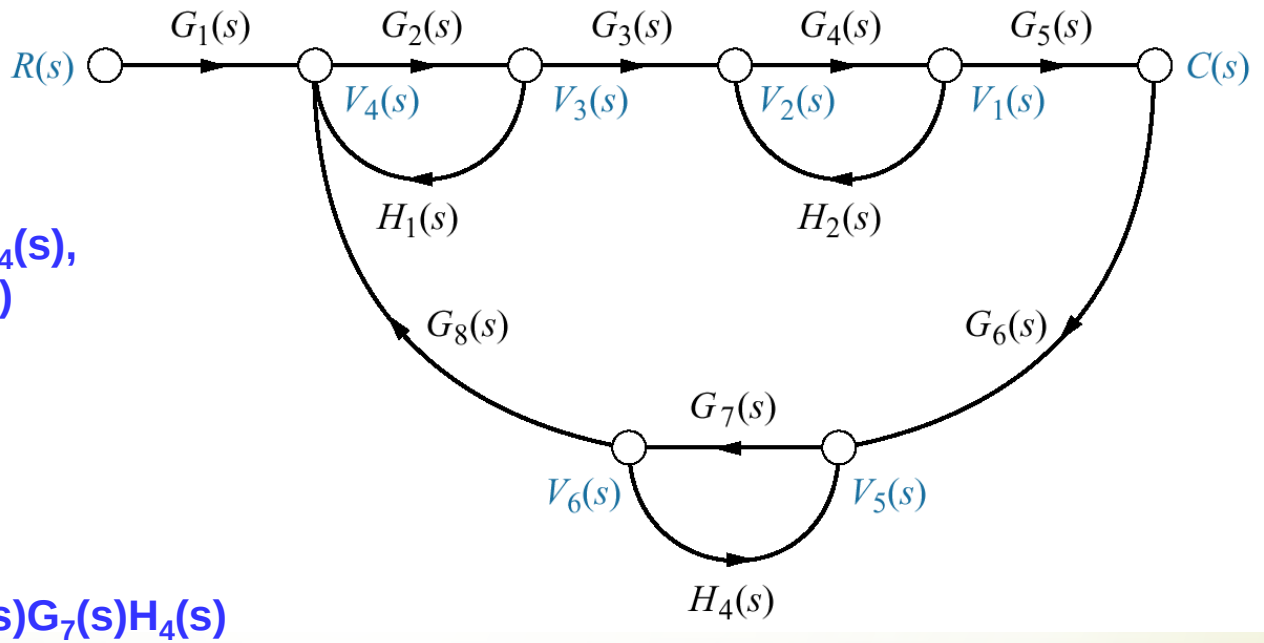
Δ = 1 - $\sum$loop gains +$\sum$ nontouching-loop gains taken 2 at a time - $\sum$ nontouching-loop gains taken 3 at a time + $\sum$ nontouching-loop gains taken 4 at a time - …….

Δ_k = 1 - loop gain terms that does not touch the k*th* forward path.

Transfer Function via Mason's Rule

Problem: Find the transfer function for the signal flow graph

Solution:



forward path

$G_1(s)G_2(s)G_3(s)G_4(s)G_5(s)$

Loop gains

$G_2(s)H_1(s)$, $G_4(s)H_2(s)$, $G_7(s)H_4(s)$,

$G_2(s)G_3(s)G_4(s)G_5(s)G_7(s)G_8(s)$

Nontouching loops

2 at a time

$G_2(s)H_1(s)G_4(s)H_2(s)$

$G_2(s)H_1(s)G_7(s)H_4(s)$

$G_4(s)H_2(s)G_7(s)H_4(s)$

3 at a time $G_2(s)H_1(s)G_4(s)H_2(s)G_7(s)H_4(s)$

Now

$$\Delta = 1-[G_2(s)H_1(s)+G_4(s)H_2(s)+G_7(s)H_4(s)+ G_2(s)G_3(s)G_4(s)G_5(s)G_7(s)G_8(s)] + [G_2(s)H_1(s)G_4(s)H_2(s) + G_2(s)H_1(s)G_7(s)H_4(s) + G_4(s)H_2(s)G_7(s)H_4(s)] - [G_2(s)H_1(s)G_4(s)H_2(s)G_7(s)H_4(s)]$$

$$\Delta_1 = 1 - G7(s)H4(s)$$

$$G(s) = \frac{P_1\Delta_1}{\Delta} = \frac{[G_1(s)G_2(s)G_3(s)G_4(s)G_5(s)][1-G_7(s)H_4(s)]}{\Delta}$$

Associate Prof. Dr . Mohamed Ahmed Ebrahim

Signal-Flow Graphs of State Equations

Problem: draw signal-flow graph for:

$$\dot{x}_1 = 2x_1 - 5x_2 + 3x_3 + 2r$$

$$\dot{x}_2 = -6x_1 - 2x_2 + 2x_3 + 5r$$

$$\dot{x}_3 = x_1 - 3x_2 - 4x_3 + 7r$$

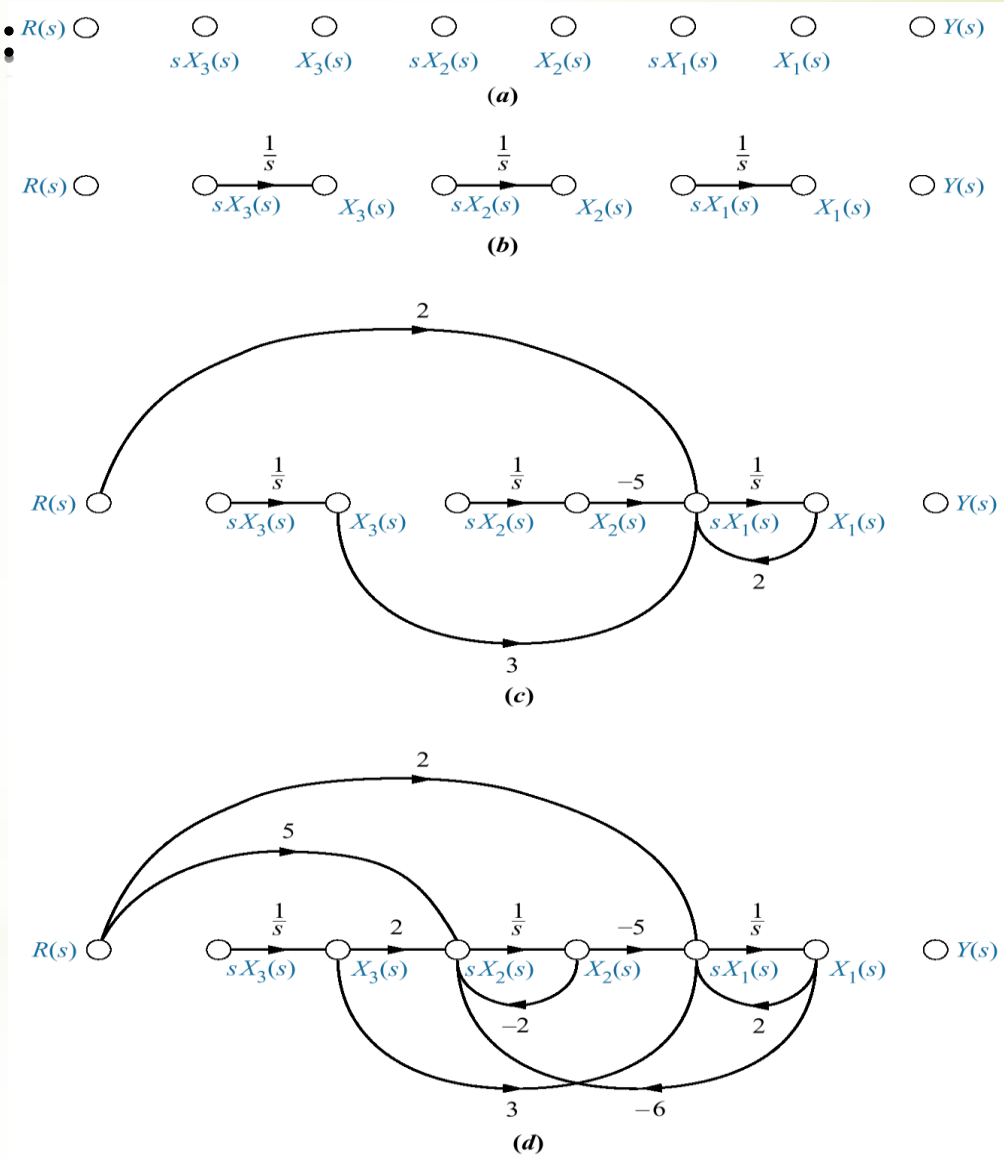
$$y = -4x_1 + 6x_2 + 9x_3$$

a. place nodes;

b. interconnect state variables and derivatives;

c. form dx1/dt ;

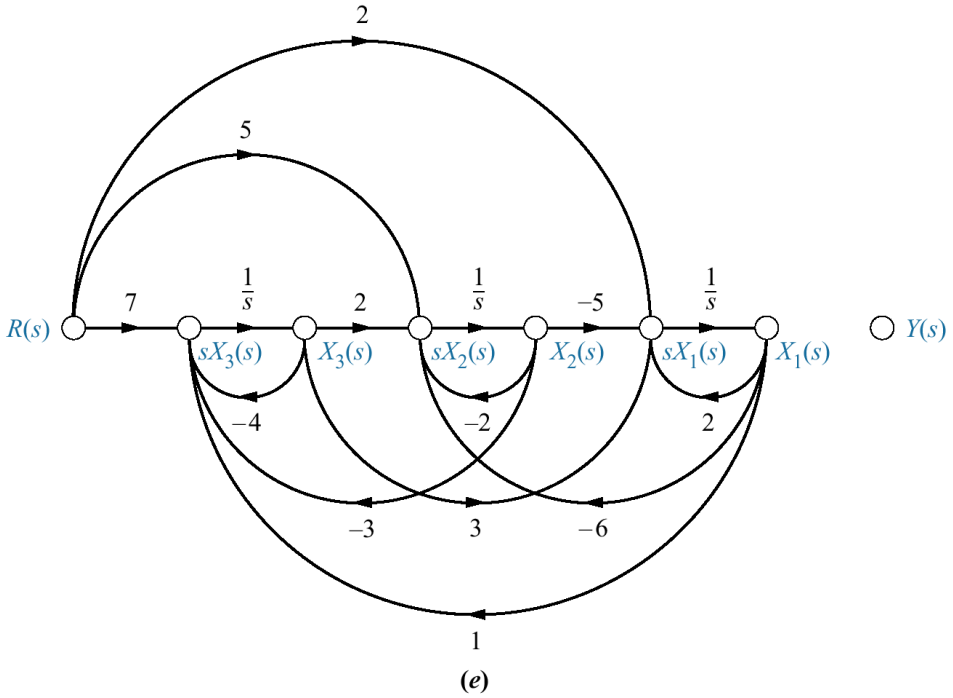
d. form dx2/dt



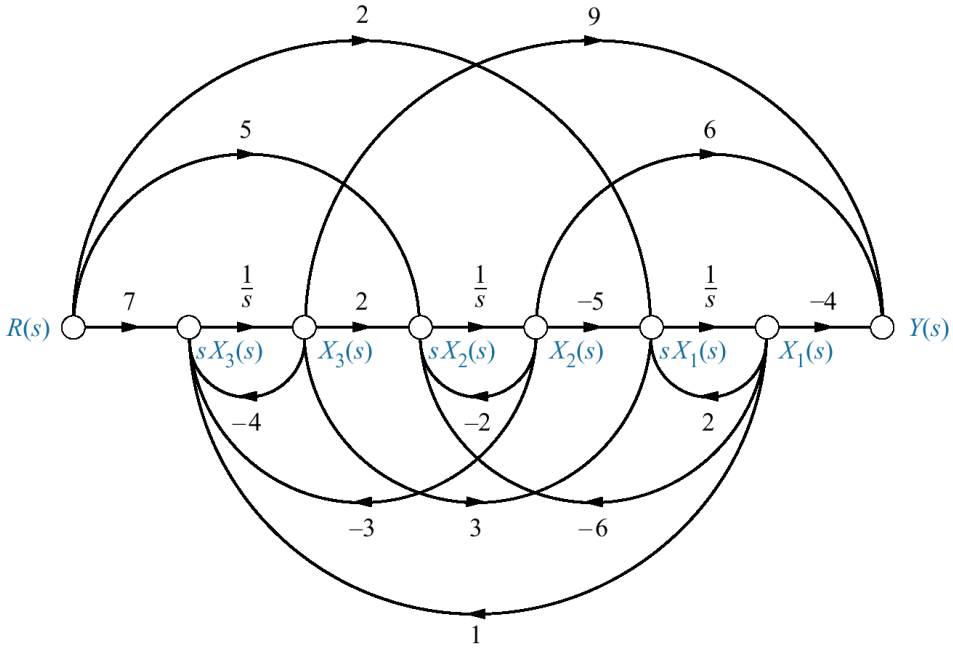
(continued)

e. form dx_3/dt;

f. form output



(e)



(f)

Mathematical Models

1- Classical or *frequency-domain* technique

2- State-Space or *Modern* or *Time-Domain* technique

Classical or Frequency-Domain Technique

- Advantages

  - Converts differential equation into algebraic equation via transfer functions.

  - Rapidly provides stability & transient response info.

- Disadvantages

  - Applicable only to Linear, Time-Invariant (LTI) systems or their close approximations.

LTI limitation became a problem circa 1960 when space applications became important.

State-Space or Modern or Time-Domain Technique

- Advantages
  - Provides a unified method for modeling, analyzing, and designing a wide range of systems using matrix algebra.
  - Nonlinear, Time-Varying, Multivariable systems

- Disadvantages
  - Not as intuitive as classical method.
  - Calculations required before physical interpretation is apparent

State-Space Representation

An LTI system is represented in state-space format by the vector-matrix differential equation (DE) as:

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) \qquad \text{Dynamic equation (s)}$$

$$\underline{y}(t) = C\underline{x}(t) + D\underline{u}(t) \qquad \text{Measurement equations}$$

with $t \geq t_0$ *and initial conditions* $\underline{x}(t_0)$.

The vectors **x, y,** and **u** are the *state*, *output* and *input vectors*.

The matrices A, B, C, and D are the *system*, *input*, *output*, and *feedforward matrices*.

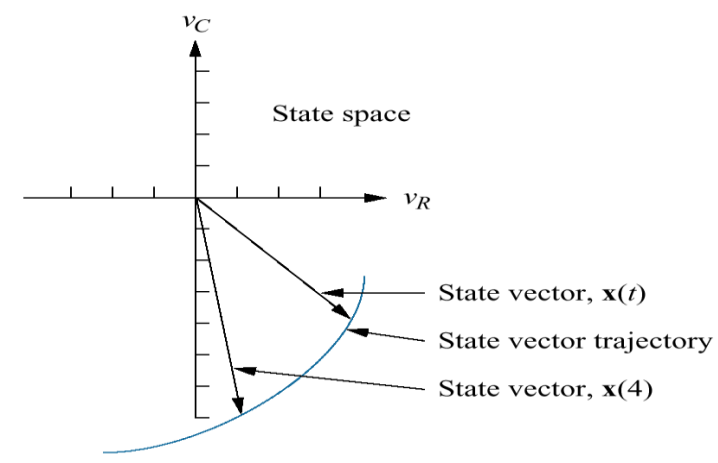
Definitions

* *System variables*: Any variable that responds to an input or initial conditions.
* *State variables:* The smallest set of *linearly independent* system variables such that the initial condition set and applied inputs *completely determine the future behavior* of the set.

Linear Independence: A set of variables is linearly independent if none of the variables can be written as a linear combination of the others.

Definitions

* *State vector:* An (n x 1) column vector whose elements are the state variables.
* *State space:* The n-dimensional space whose axes are the state variables.



Graphic representation of state space and a state vector

The minimum number of state variables is equal to:

- The order of the DE's describing the system.
- The order of the denominator polynomial of its transfer function model.
- The number of independent energy storage elements in the system.

Remember the state variables must be linearly independent! If not, you may not be able to solve for all the other system variables, or even write the state equations.

Dynamic equation

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t)$$

State equation

$$y(t) = Cx(t) + Du(t)$$

Output equation

State variable

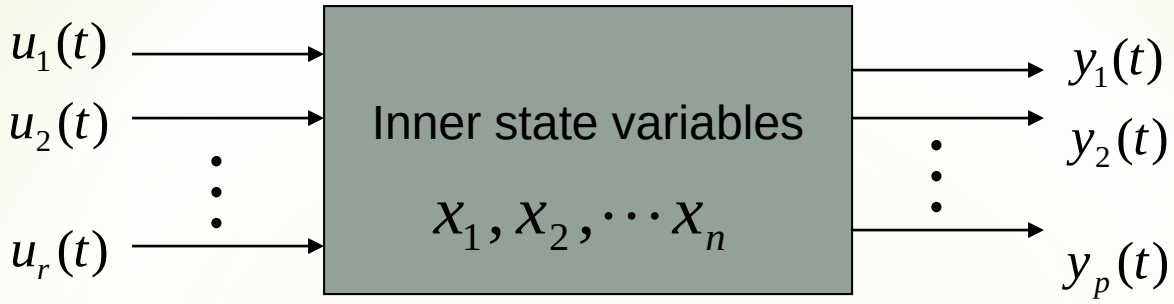
$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix}_{n\times 1} \qquad u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_r(t) \end{bmatrix}_{r\times 1} \qquad y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_p(t) \end{bmatrix}_{p\times 1} \qquad x(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \\ \vdots \\ x_n(0) \end{bmatrix}_{n\times 1}$$

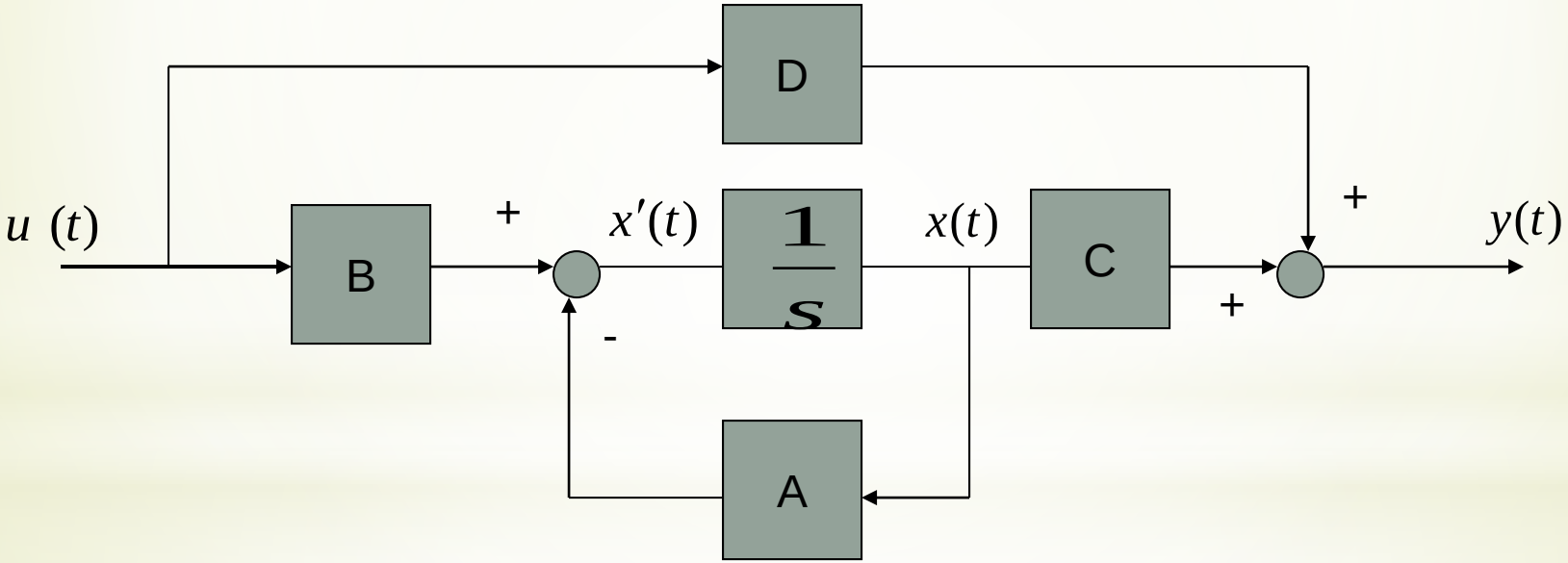
State space

r- input

p- output

$$A = \begin{bmatrix} n\times n \end{bmatrix} \qquad B = \begin{bmatrix} n\times r \end{bmatrix} \qquad C = \begin{bmatrix} p\times n \end{bmatrix} \qquad D = \begin{bmatrix} p\times r \end{bmatrix}$$


$u_1(t)$
$u_2(t)$
$u_r(t)$
Inner state variables
$x_1, x_2, \cdots x_n$
$y_1(t)$
$y_2(t)$
$y_p(t)$


D
$u\ (t)$
B
+
$x'(t)$
$\frac{1}{s}$
$x(t)$
C
+
+
$y(t)$
-
A

Dynamical equation → Transfer function

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t) + Du(t)$$

Laplace transform

$$sX(s) - x(0) = AX(s) + BU(s)$$
$$Y(s) = CX(s) + DU(s)$$

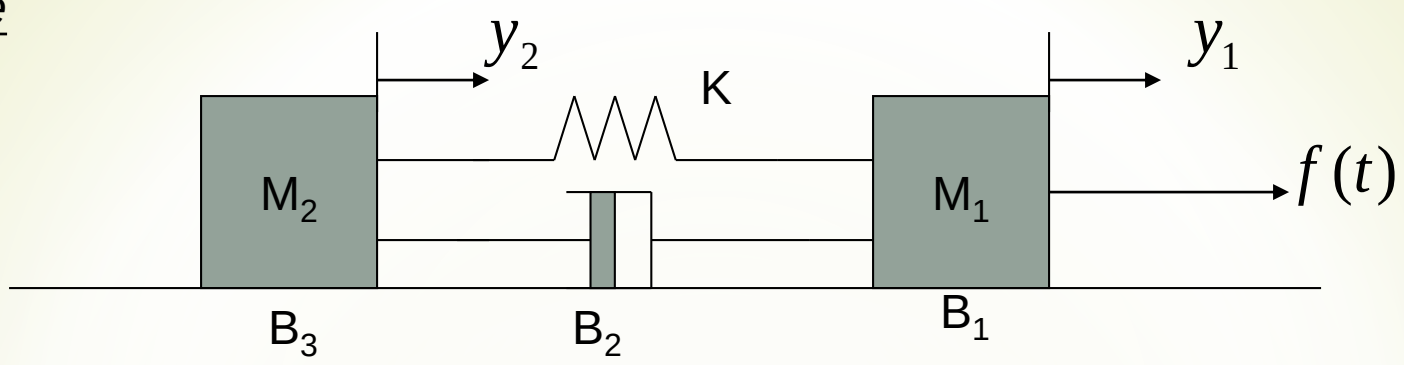
$$assume \quad x(0) = 0$$

$$X(s) = (sI - A)^{-1} BU(s)$$
$$Y(s) = [C(sI - A)^{-1} B + D]U(s)$$

Transfer function

matrix

Example

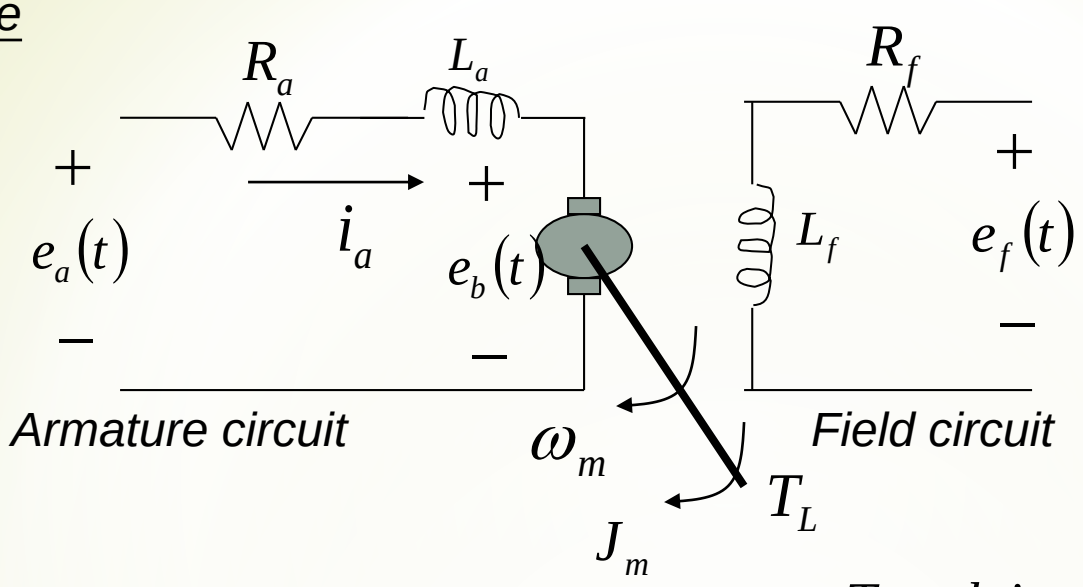


$$M_1\ddot{y}_1 + B_1\dot{y}_1 + B_2(\dot{y}_1 - \dot{y}_2) + K(y_1 - y_2) = f(t)$$

$$M_2\ddot{y}_2 + B_3\dot{y}_2 + B_2(\dot{y}_2 - \dot{y}_1) + K(y_2 - y_1) = 0$$

$$let \begin{cases} x_1 = y_1 - y_2 \\ x_2 = \dot{y}_1 \\ x_3 = \dot{y}_2 \end{cases} \Rightarrow \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & -1 \\ -\frac{K}{M_1} & -\frac{(B_1+B_2)}{M_1} & \frac{B_2}{M_1} \\ \frac{K}{M_2} & \frac{B_2}{M_2} & -\frac{(B_2+B_3)}{M_2} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M_1} \\ 0 \end{bmatrix} f(t)$$

Example



$$\begin{cases} e_a(t) = R_a i_a + e_b + L_a \dfrac{di_a}{dt} \\ e_b = k_b \omega_m \end{cases}$$

$$\frac{di_a}{dt} = \frac{-R_a}{L_a} i_a - \frac{k_b}{L_a} \omega_m + \frac{1}{L_a} e_a$$

$$T_m = k_i i_a$$

$$= J_m \frac{d^2\theta_m}{dt^2} + B_m \frac{d\theta_m}{dt} + T_L$$

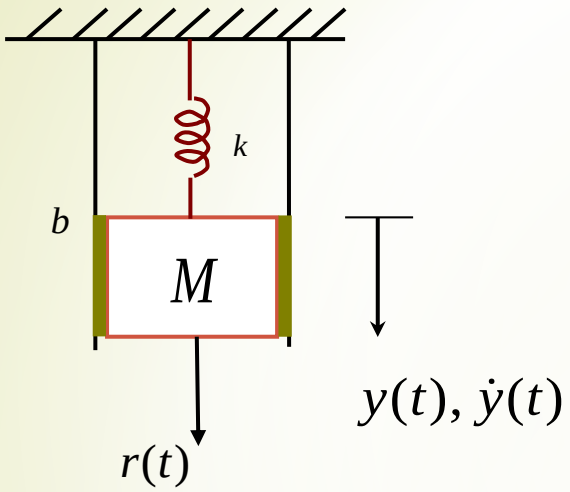
$$\frac{d\omega_m}{dt} = \frac{k_i}{J_m} i_a - \frac{B_m}{J_m} \frac{d\theta_m}{dt} - \frac{1}{J_m} T_L$$

$$\frac{d\theta_m}{dt} = \omega_m$$

$$\begin{bmatrix} \dot{i}_a \\ \dot{\omega}_m \\ \dot{\theta}_m \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & \frac{-K_b}{L_a} & 0 \\ \frac{K_i}{J_m} & \frac{-B_m}{J_m} & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_a \\ \omega_m \\ \theta_m \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J_m} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e_a \\ T_L \end{bmatrix}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \end{bmatrix} = \begin{bmatrix} \theta_m(t) \\ \omega_m(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} i_a \\ \omega_m \\ \theta_m \end{bmatrix}$$

Example



By Newton's Law

$$F = M\ddot{y}$$

$$r - ky - b\dot{y} = M\ddot{y}$$

$$\Rightarrow M\ddot{y} + b\dot{y} + ky = r$$

let $x_1 = y, x_2 = \dot{y}$

$$\Rightarrow \begin{cases} \dot{x}_1 = \dot{y} = x_2 \\ \dot{x}_2 = \ddot{y} = -\frac{b}{M}\dot{y} - \frac{k}{M}y + \frac{1}{M}r \end{cases}$$

$$= -\frac{b}{M}x_2 - \frac{k}{M}x_1 + \frac{1}{M}u \qquad (u = r)$$

$$\Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{k}{M}x_1 - \frac{b}{M}x_2 + \frac{1}{M}u \end{cases}$$

r: external force

k: spring

b: coefficient of viscous friction

$$\Rightarrow \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{b}{M} \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix}}_{B} \cdot u$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0 \cdot u$$

State Space Equation

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

$$\text{For example}: \begin{cases} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{M} & -\frac{b}{M} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} u \\ y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0 \cdot u \end{cases}$$

Transfer Function

$$G(s) = \frac{Y(s)}{U(s)}$$

$$\text{For example}: G(s) = \frac{b_1 s + b_0}{a_2 s^2 + a_1 s + a_0}$$

Example: Transfer function of the Mass-damper-spring system

$$M\frac{d^2y}{dt^2} + b\frac{dy}{dt} + ky = u(t)$$

$$x = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T$$

$$Ms^2Y(s) + bsY(s) + kY(s) = U(s)$$

$$\frac{Y(s)}{U(s)} = G(s) = \frac{1}{Ms^2 + bs + k}$$

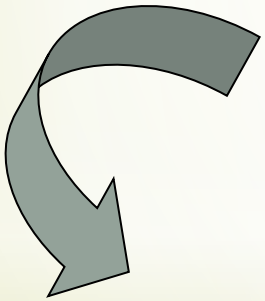
Example

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0 & 1 & 0\\ 0 & -4 & 3\\ -1 & -1 & -2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}0 & 0\\ 1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}u_1\\ u_2\end{bmatrix}$$

MIMO system

$$\begin{bmatrix}y_1(t)\\ y_2(t)\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}$$

$$(sI-A)^{-1}=\frac{adj(sI-A)}{|sI-A|}$$



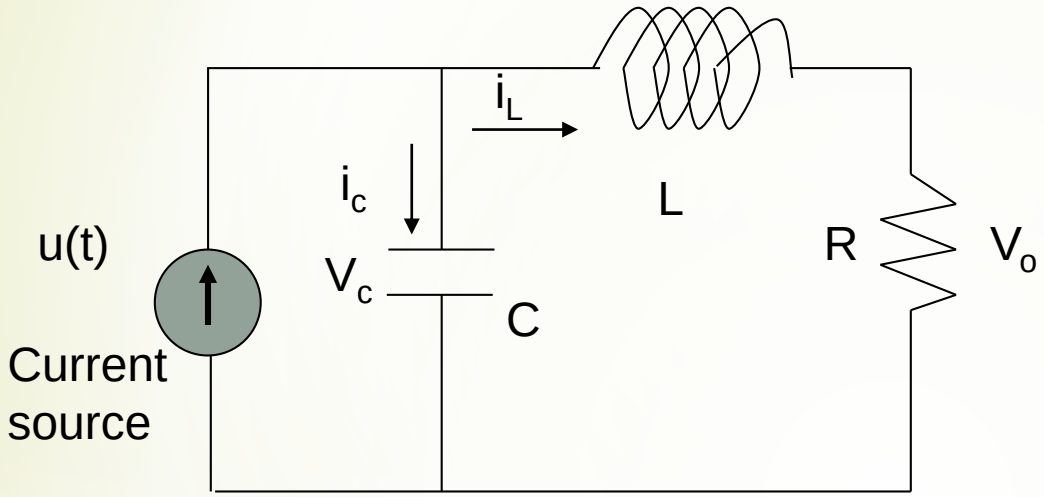
$$=\frac{1}{s(s+4)(s+2)+3+3s}\begin{bmatrix}s^2+6s+11 & s+2 & 3\\ -3 & s^2+2 & 3s\\ s+4 & -s-1 & s^2+4s\end{bmatrix}$$

$$G(s)=[C(sI-A)^{-1}B+D]$$

$$=\frac{1}{s^3+6s^2+11s+3}\begin{bmatrix}s+2 & 3\\ -(s+1) & s(s+4)\end{bmatrix}$$

Transfer function

As another example of the state variable characterization of a system, consider the RLC circuit shown in the following figure.



The state of this system can be described in terms of a set of variables [x_1 x_2], where x_1 is the capacitor voltage $v_c(t)$ and x_2 is equal to the inductor current $i_L(t)$. This choice of state variables is intuitively satisfactory because the stored energy of the network can be described in terms of these variables.

$$E_1 = \frac{1}{2} L\, i_L{}^2, \quad E_2 = \frac{1}{2C}\left(\int i_c dt\right)^2 = \frac{1}{2} C\, v_c^2$$

Therefore $x_1(t_0)$ and $x_2(t_0)$ represent the total initial energy of the network and thus the state of the system at $t=t_0$.

Utilizing Kirchhoff's current low at the junction, we obtain a first order differential equation by describing the rate of change of capacitor voltage

$$i_c = C\frac{dv_c}{dt} = u(t) - i_L$$

Kirchhoff's voltage low for the right-hand loop provides the equation describing the rate of change of inducator current as

$$L\frac{di_L}{dt} = -R\,i_L + v_c$$

The output of the system is represented by the linear algebraic equation

$$v_0 = R\,i_L(t)$$

We can write the equations as a set of two first order differential equations in terms of the state variables x_1 [$v_C(t)$] and x_2 [$i_L(t)$] as follows:

$$C\frac{dv_c}{dt} = u(t) - i_L \quad \Longrightarrow \quad \frac{dx_1}{dt} = -\frac{1}{C}x_2 + \frac{1}{C}u(t)$$

$$L\frac{di_L}{dt} = -R\,i_L + v_c \quad \Longrightarrow \quad \frac{dx_2}{dt} = \frac{1}{L}x_1 - \frac{R}{L}x_2$$

The output signal is then $y_1(t) = v_0(t) = R\,x_2$

Utilizing the first-order differential equations and the initial conditions of the network represented by [$x_1(t_0)$ $x_2(t_0)$], we can determine the system's future and its output.

The state variables that describe a system are not a unique set, and several alternative sets of state variables can be chosen. For the RLC circuit, we might choose the set of state variables as the two voltages, $v_C(t)$ and $v_L(t)$.

In an actual system, there are several choices of a set of state variables that specify the *energy stored in a system* and therefore adequately describe the dynamics of the system.

The state variables of a system characterize the dynamic behavior of a system. The engineer's interest is primarily in physical, where the variables are voltages, currents, velocities, positions, pressures, temperatures, and similar physical variables.

The State Differential Equation:

The state of a system is described by the set of first-order differential equations written in terms of the state variables [x_1 x_2 ... x_n]. These first-order differential equations can be written in general form as

$$\dot{x}_1 = a_{11}x_1 + a_{12}x_2 + \ldots a_{1n}x_n + b_{11}u_1 + \cdots b_{1m}u_m$$
$$\dot{x}_2 = a_{21}x_1 + a_{22}x_2 + \ldots a_{2n}x_n + b_{21}u_1 + \cdots b_{2m}u_m$$
$$\vdots$$
$$\dot{x}_n = a_{n1}x_1 + a_{n2}x_2 + \ldots a_{nn}x_n + b_{n1}u_1 + \cdots b_{nm}u_m$$

Thus, this set of simultaneous differential equations can be written in matrix form as follows:

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \cdots & \cdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & \cdots & b_{1m} \\ \vdots & \cdots & \vdots \\ b_{n1} & \cdots & b_{nm} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

n: number of state variables, m: number of inputs.

The column matrix consisting of the state variables is called the **state vector** and is written as

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

The vector of input signals is defined as u. Then the system can be represented by the compact notation of the state differential equation as

$$\dot{x} = A\,x + B\,u$$

This differential equation is also commonly called the state equation. The matrix **A** is an nxn square matrix, and **B** is an nxm matrix. The state differential equation relates the rate of change of the state of the system to the state of the system and the input signals. In general, the outputs of a linear system can be related to the state variables and the input signals by the output equation

$$y = C\,x + D\,u$$

Where **y** is the set of output signals expressed in column vector form. The state-space representation (or state-variable representation) is comprised of the state variable differential equation and the output equation.

We can write the state variable differential equation for the RLC circuit as

$$\dot{x} = \begin{bmatrix} 0 & -\frac{1}{C} \\ \frac{1}{L} & -\frac{R}{L} \end{bmatrix} x + \begin{bmatrix} \frac{1}{C} \\ 0 \end{bmatrix} u(t)$$

and the output as

$$y = \begin{bmatrix} 0 & R \end{bmatrix} x$$

The solution of the state differential equation can be obtained in a manner similar to the approach we utilize for solving a first order differential equation. Consider the first-order differential equation

$$\dot{x} = ax + bu$$

Where x(t) and u(t) are scalar functions of time. We expect an exponential solution of the form $e^{at.}$ Taking the Laplace transform of both sides, we have

$$s\,X(s) - x_0 = a\,X(s) + b\,U(s)$$

therefore,

$$X(s) = \frac{x(0)}{s-a} + \frac{b}{s-a}\,U(s)$$

The inverse Laplace transform of X(s) results in the solution

$$x(t) = e^{at}\,x(0) + \int_0^t e^{a(t-\tau)}\,b\,u(\tau)\,d\tau$$

We expect the solution of the state differential equation to be similar to x(t) and to be of differential form. The **matrix exponential function** is defined as

$$e^{At} = I + At + \frac{A^2t^2}{2!} + \cdots + \frac{A^kt^k}{k!} + \cdots$$

which converges for all finite t and any A. Then the solution of the state differential equation is found to be

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}B\,u(\tau)\,d\tau$$

$$X(s) = \left[sI - A\right]^{-1}x(0) + \left[sI - A\right]^{-1}B\,U(s)$$

where we note that $[sI-A]^{-1}=\phi(s)$, which is the Laplace transform of $\phi(t)=e^{At}$. The matrix exponential function $\phi(t)$ describes the unforced response of the system and is called the fundamental or state transition matrix.

$$x(t) = \phi(t)\,x(0) + \int_0^t \phi(t-\tau)\,B\,u(\tau)\,d\tau$$

THE TRANSFER FUNCTION FROM THE STATE EQUATION

The transfer function of a single input-single output (SISO) system can be obtained from the state variable equations.

$$\dot{x} = A\,x + B\,u$$
$$y = C\,x$$

where y is the single output and u is the single input. The Laplace transform of the equations

$$sX(s) = AX(s) + B\,U(s)$$
$$Y(s) = CX(s)$$

where B is an nx1 matrix, since u is a single input. We do not include initial conditions, since we seek the transfer function. Reordering the equation

$$[sI - A]X(s) = B\,U(s)$$

$$X(s) = [sI - A]^{-1}BU(s) = \phi(s)BU(s)$$

$$Y(s) = C\phi(s)BU(s)$$

Therefore, the transfer function G(s)=Y(s)/U(s) is

$$G(s) = C\phi(s)B$$

Example:

Determine the transfer function G(s)=Y(s)/U(s) for the RLC circuit as described by the state differential function

$$\dot{x} = \begin{bmatrix} 0 & -\frac{1}{C} \\ \frac{1}{L} & -\frac{R}{L} \end{bmatrix} x + \begin{bmatrix} \frac{1}{C} \\ 0 \end{bmatrix} u \quad , \quad y = \begin{bmatrix} 0 & R \end{bmatrix} x$$

$$[sI-A]=\begin{bmatrix} s & \frac{1}{C} \\ -\frac{1}{L} & s+\frac{R}{L} \end{bmatrix}$$

$$\phi(s)=[sI-A]^{-1}=\frac{1}{\Delta(s)}\begin{bmatrix} s+\frac{R}{L} & -\frac{1}{C} \\ \frac{1}{L} & s \end{bmatrix}$$

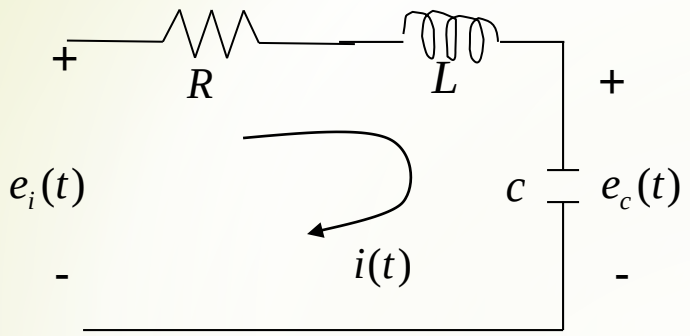
$$\Delta(s)=s^2+\frac{R}{L}s+\frac{1}{LC}$$

Then the transfer function is

$$G(s)=\begin{bmatrix}0 & R\end{bmatrix}\begin{bmatrix} \frac{s+\frac{R}{L}}{\Delta(s)} & -\frac{1}{C\Delta(s)} \\ \frac{1}{L\Delta(s)} & \frac{s}{\Delta(s)} \end{bmatrix}\begin{bmatrix}\frac{1}{C}\\ 0\end{bmatrix}$$

$$G(s)=\frac{R/LC}{\Delta(s)}=\frac{R/LC}{s^2+\frac{R}{L}s+\frac{1}{LC}}$$

Remark : the choice of states is not unique.



$$Ri(t) + L\frac{di(t)}{dt} + \frac{1}{c}\int_0^t i(t)dt = e_i(t)$$

let

$$x_1(t) = i(t)$$
$$x_2(t) = \int i(t)dt$$
$$y(t) = i(t)$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}-\frac{R}{L} & -\frac{1}{LC}\\ 1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}\frac{1}{L}\\ 0\end{bmatrix}e_i(t)$$

$$y(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

let

$$\hat{x}_1(t) = i(t)$$
$$\hat{x}_2(t) = e_c(t)$$
$$y(t) = i(t)$$

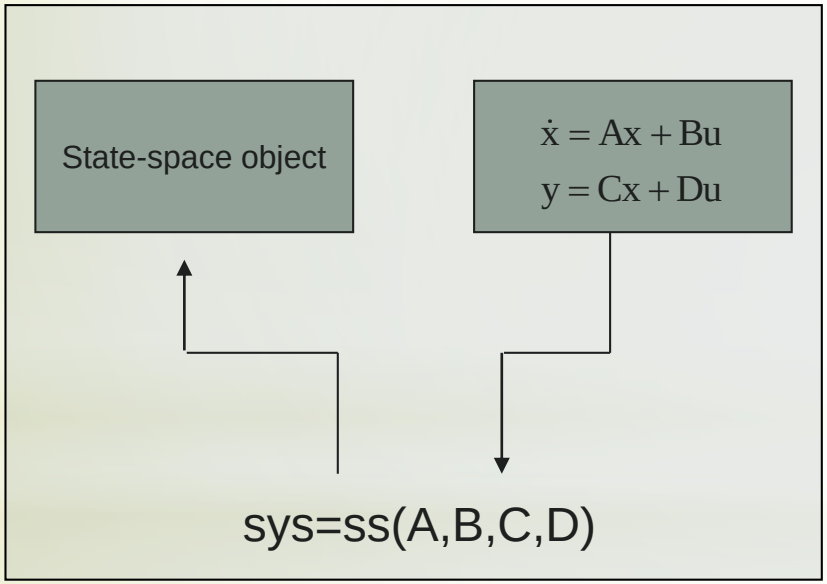
exist a mapping

$$\begin{bmatrix}\dot{\hat{x}}_1\\ \dot{\hat{x}}_2\end{bmatrix} = \begin{bmatrix}-\frac{R}{L} & -\frac{R}{L}\\ \frac{1}{L} & 0\end{bmatrix}\begin{bmatrix}\hat{x}_1\\ \hat{x}_2\end{bmatrix} + \begin{bmatrix}\frac{1}{L}\\ 0\end{bmatrix}e_i(t)$$

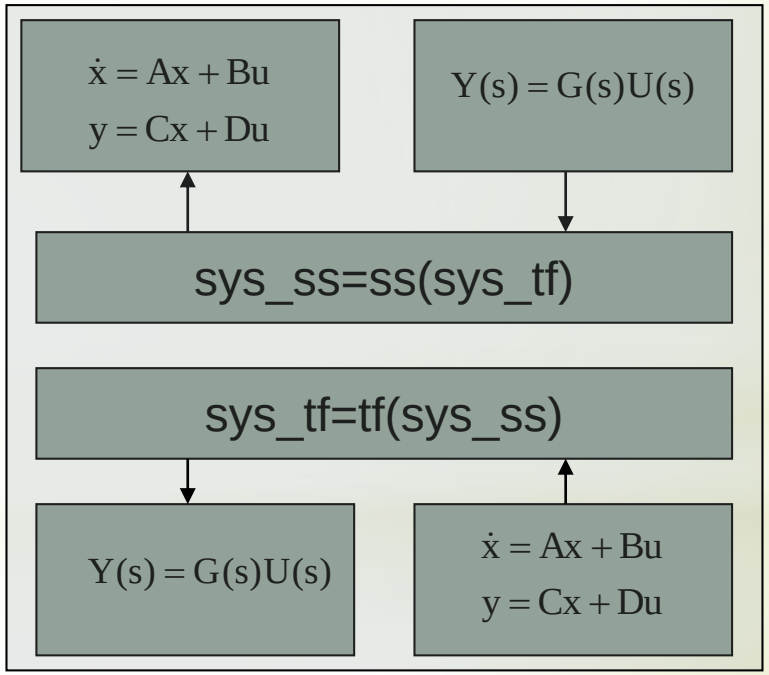
$$y(t) = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\hat{x}_1\\ \hat{x}_2\end{bmatrix}$$

ANALYSIS OF STATE VARIABLE MODELS USING MATLAB

Given a transfer function, we can obtain an equivalent state-space representation and vice versa. The function **tf** can be used to convert a state-space representation to a transfer function representation; the function **ss** can be used to convert a transfer function representation to a state-space representation. The functions are shown in Figure 4, where sys_tf represents a transfer function model and sys_ss is a state space representation.



The **ss** function



Linear system model conversion

For instance, consider the third-order system

$$G(s) = \frac{Y(s)}{R(s)} = \frac{2s^2 + 8s + 6}{s^3 + 8s^2 + 16s + 6}$$

We can obtain a state-space representation using the **ss** function. The state-space representation of the system given by G(s) is

Matlab code

```
num=[2 8 6];den=[1 8 16 6];
sys_tf=tf(num,den)
sys_ss=ss(sys_tf)
```

Answer

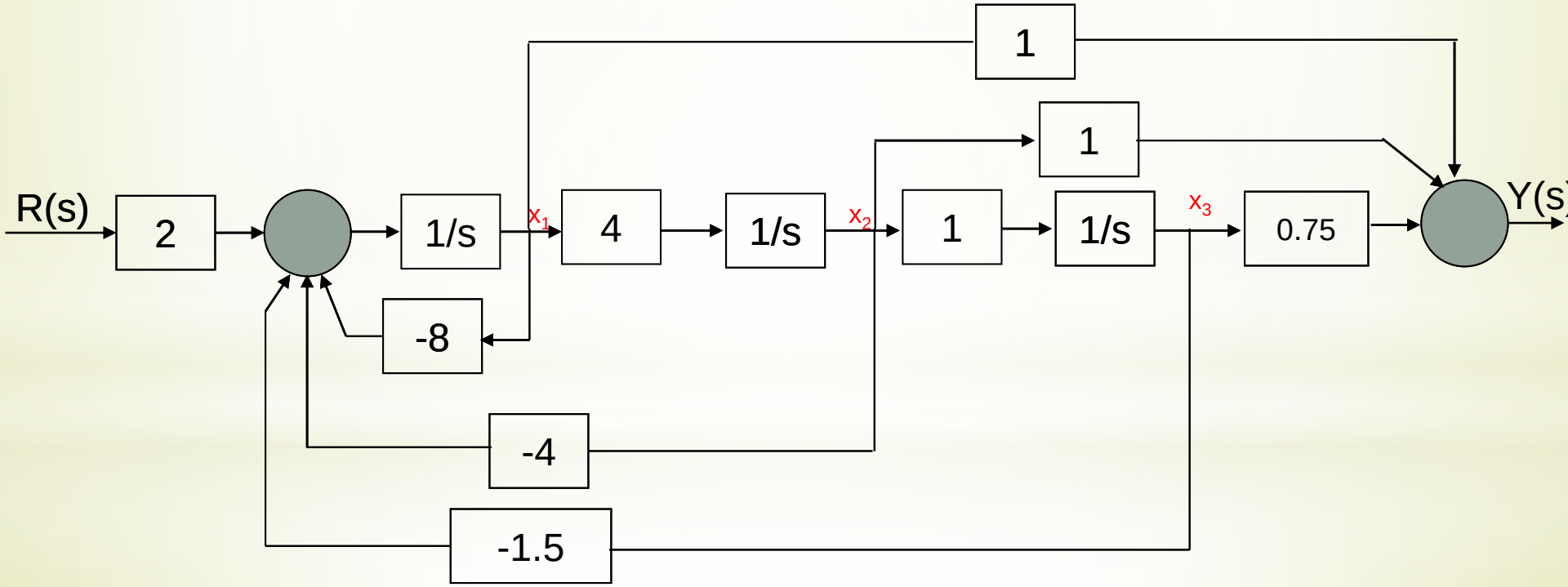
```
Transfer function:
  2 s^2 + 8 s + 6
----------------------
s^3 + 8 s^2 + 16 s + 6


a =
       x1   x2   x3
  x1   -8   -4 -1.5
  x2    4    0    0
  x3    0    1    0

b =
     u1
  x1  2
  x2  0
  x3  0

c =
       x1   x2   x3
  y1    1    1 0.75

d =
     u1
  y1  0

Continuous-time model.
```

$$A = \begin{bmatrix} -8 & -4 & -1.5 \\ 4 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 0.75 \end{bmatrix} \text{ and } D = \begin{bmatrix} 0 \end{bmatrix}$$

$$A = \begin{bmatrix} -8 & -4 & -1.5 \\ 4 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, B = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 1 & 0.75 \end{bmatrix} \text{ and } D = \begin{bmatrix} 0 \end{bmatrix}$$



Block diagram with x_1 defined as the leftmost state variable.

$$x(t) = e^{At}x(0) + \int_0^t e^{A(t-\tau)}B\,u(\tau)\,d\tau$$

$$x(t) = \phi(t)\,x(0) + \int_0^t \phi(t-\tau)\,B\,u(\tau)\,d\tau$$

We can use the function **expm** to compute the transition matrix for a given time. The **expm(A)** function computes the matrix exponential. By contrast the exp(A) function calculates $e^{a_{ij}}$ for each of the elements a_{ij}ϵA.

For the RLC network, the state-space representation is given as:

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

The initial conditions are $x_1(0)=x_2(0)=1$ and the input u(t)=0. At t=0.2, the state transition matrix is calculated as

>>A=[0 -2;1 -3], dt=0.2; Phi=expm(A*dt)

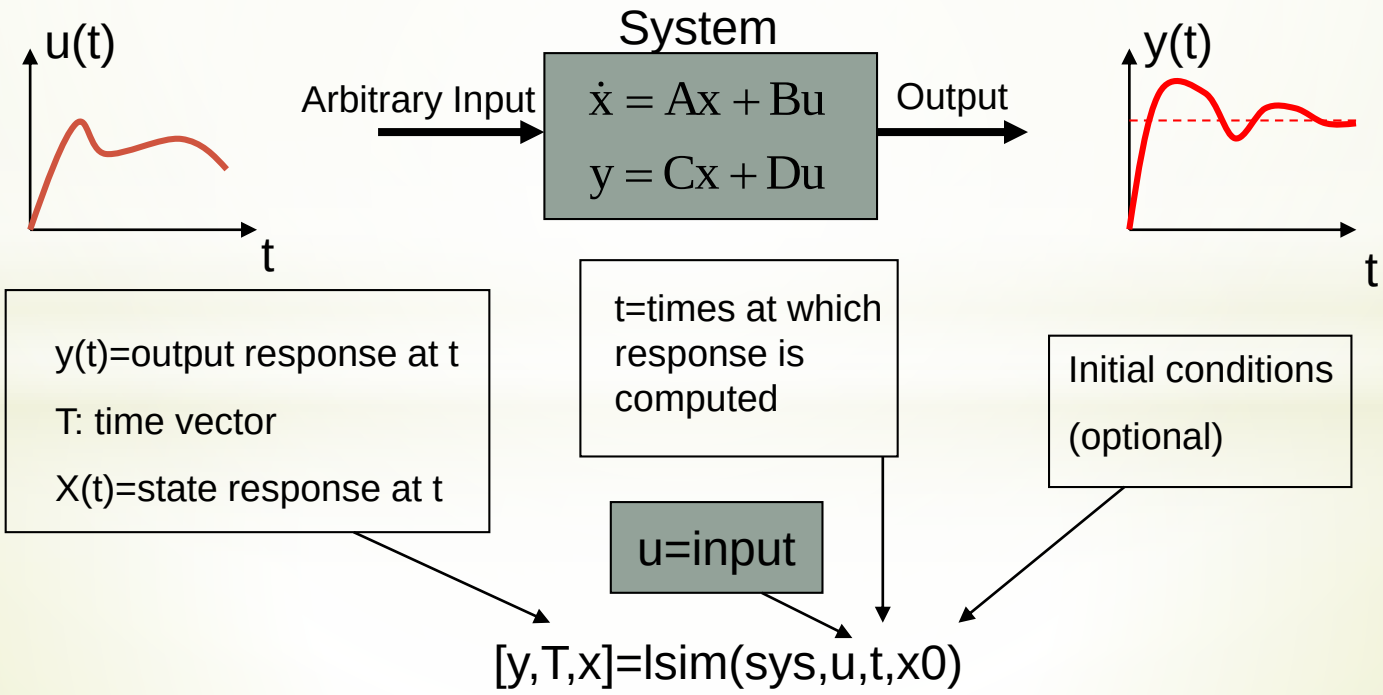
Phi =

0.9671 -0.2968
0.1484 0.5219

The state at t=0.2 is predicted by the state transition method to be

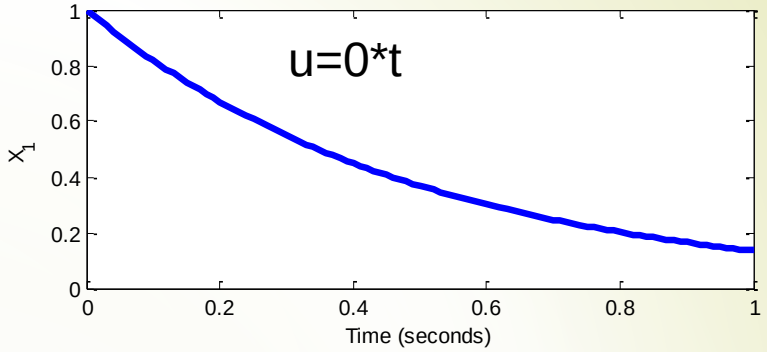
$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t=0.2} = \begin{bmatrix} 0.9671 & -0.2968 \\ 0.1484 & 0.5219 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t=0} = \begin{bmatrix} 0.6703 \\ 0.6703 \end{bmatrix}$$

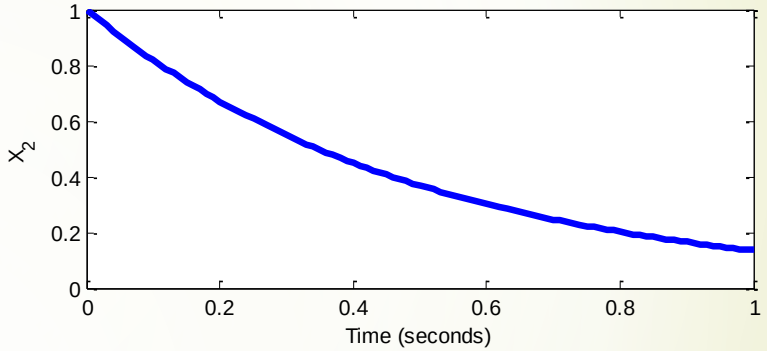
The time response of a system can also be obtained by using **lsim** function. The **lsim** function can accept as input nonzero initial conditions as well as an input function. Using **lsim** function, we can calculate the response for the RLC network as shown below.

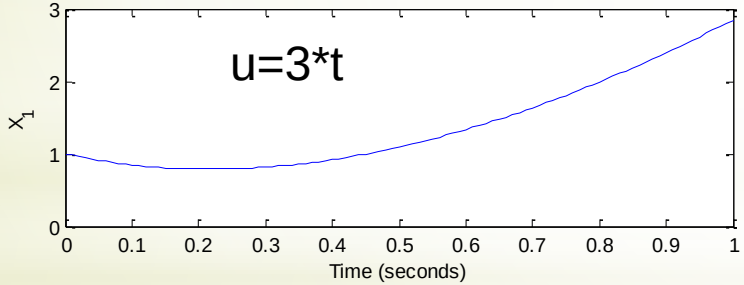


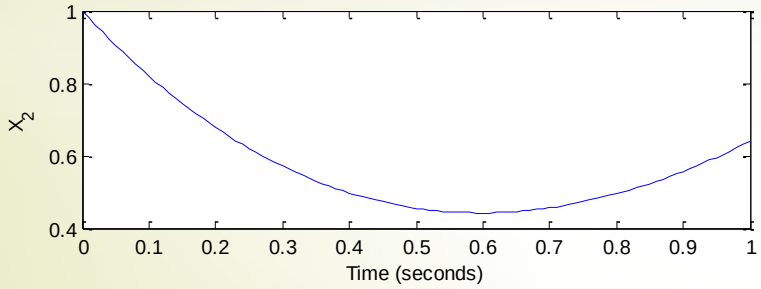
Matlab code

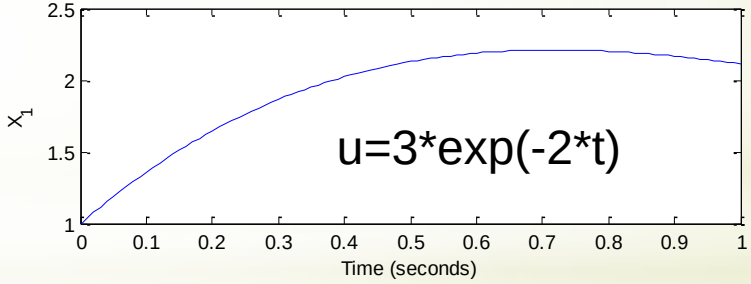
```
clc;clear
A=[0 -2;1 -3];B=[2;0];C=[1 0];D=[0];
sys=ss(A,B,C,D)   %state-space model
x0=[1 1]; %initial conditions
t=[0:0.01:1];
u=0*t;  %zero input
[y,T,x]=lsim(sys,u,t,x0);
subplot(211),plot(T,x(:,1))
xlabel('Time (seconds)'),ylabel('X_1')
subplot(212),plot(T,x(:,2))
xlabel('Time (seconds)'),ylabel('X_2')
```

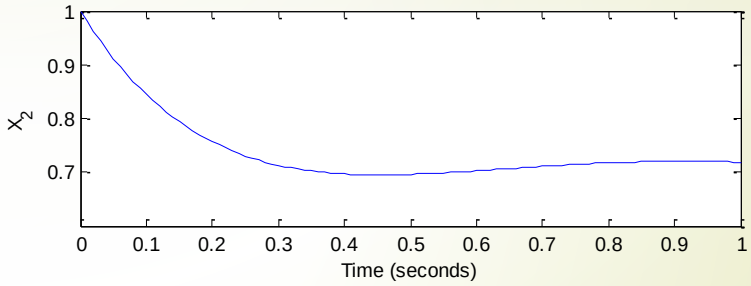












With Our Best Wishes

Automatic Control (1)

Course Staff

Associate Prof. Dr . Mohamed Ahmed Ebrahim

Thank You
For Your Attention

Mohamed Ahmed Ebrahim

Associate Prof. Dr . Mohamed Ahmed Ebrahim