Notebook

Control

Sec
Semester
2024

$x(t) = L^{-1}(X(s))$

$f(t) \xleftarrow[L^{-1}]{} F(s)$

The Laplace Transform

$f(t)$	$F(s)$
$\delta(t)$	1
$u(t) =$ الثابت	$\frac{\text{الثابت}}{s}$
e^{-at}	$\frac{1}{s+a}$
t	$\frac{1}{s^2}$
t^n	$\frac{n!}{s^{n+1}}$
te^{-at}	$\frac{1}{(s+a)^2}$
$t^n e^{-at}$	$\frac{n!}{(s+a)^{n+1}}$
$\sin(wt)$	$\frac{w}{s^2+w^2}$
$\cos(wt)$	$\frac{s}{s^2+w^2}$
$e^{-at}\sin(wt)$	$\frac{w}{(s+a)^2+w^2}$
$e^{-at}\cos(wt)$	$\frac{s+a}{(s+a)^2+w^2}$
$\sin(wt+\theta)$	$\frac{s\sin\theta + w\cos\theta}{s^2+w^2}$
$\cos(wt+\theta)$	$\frac{s\cos\theta - w\sin\theta}{s^2+w^2}$

*** Initial Value Theorem :-**

$$\lim_{s\to\infty} sF(s) = \lim_{t\to 0} f(t) = f(0)$$

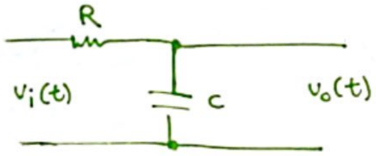
*** Final Value Theorem :-**

$$\lim_{s\to 0} sF(s) = \lim_{t\to\infty} f(t) = f(\infty)$$

Basic Elements of Electrical Systems

(V-I) Relation

Resistor	$V_R(t) = i_R(t) \times R$
Capacitor	$V_c(t) = \frac{1}{C} \int i_c(t)\,dt$
Inductor	$V_L(t) = L \frac{di_L(t)}{dt}$



Transfer function →

$$\boxed{\frac{V_o(s)}{V_i(s)} = \frac{1}{1+RCs}}$$

→ one pole
جذر واحد للمقام
$\left(s = -\frac{1}{RC}\right)$

impedence form
$Z_R \rightarrow R$
$Z_C \rightarrow \frac{1}{Cs}$
$Z_L \rightarrow Ls$

Basic Types of Mechanical systems

Translational

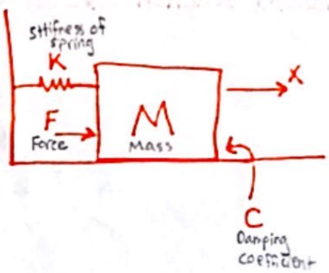
Translational → Linear motion
Rotational → Rotation motion

* Notes:-
Spring → resist the translatio
Mass → resist the acceleration
Resistor → resist the flow of curr
capacitor → resist the flow of curren

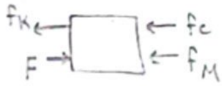
*Basic Types of Translational Mechanical Systems
- Spring → $F = \underset{\text{stiffness of spring } (N/m)}{K}(x_1 - x_2)$
- Mass → $F = M\ddot{x}$
- Damper → $F = \underset{\text{damping coefficient } \left(\frac{N}{ms}\right)}{C}(\dot{x}_1 - \dot{x}_2)$

* Notes:-
Position K → تحاول تقاوم تغيير الـ position عن طريق (K)x(X)
Mass → يقاوم الـ Force عن طريق (mass)x(acceleration) $m \times \ddot{x}$
Damper → يقاوم عن طريق $(C) \times (\dot{x})$

*consider the following system



⟹ . Free Body Diagram:



$F_{friction} = \mu N$

$$F = \overset{spring}{f_K} + \overset{mass}{f_M} + \overset{damper}{f_c}$$

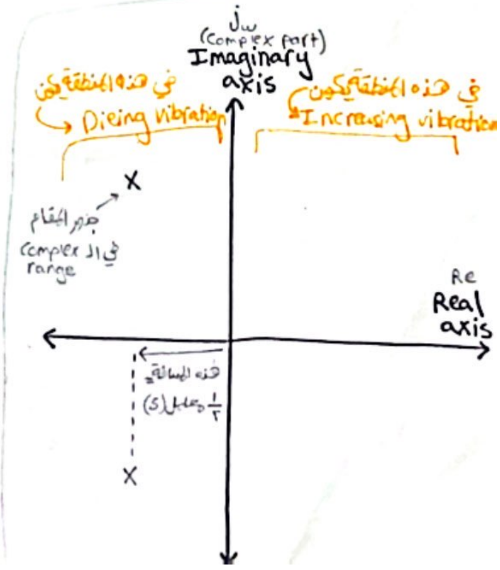
$$F = Kx + M\underset{\text{acceleration}}{\ddot{x}} + C\underset{\text{velocity}}{\dot{x}}$$

Now by Taking the Laplace Transform $\Rightarrow F_{(s)} = K\,x(s) + Ms^2X(s) + Cs\,X(s)$

الان سوف اخذ X(s) عامل مشترك
و احسب الـ Transfer function
$\frac{X(s)}{F(s)}$

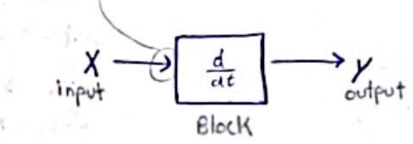
$$\frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Cs + K}$$

* اذا اجى الجذر على الـ imaginary axis ← sinusoidal
* اذا اجى الجذر على الـ Real axis ← exponential
* اذا اجى الجذر مكرر على الـ Real axis ← t x exponential

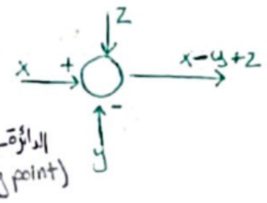


Block Diagram

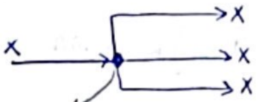
the Arrow represent the direction of information flow



البوكس يمثل عملية ضرب

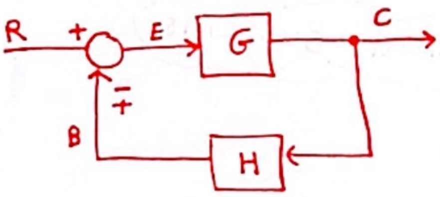


الدائرة ← هي عملية جمع
(summing point)



Takeoff point
or (pickoff point)

* Canonical Form of a Feedback Control System :-



control Ratio (closed loop transfer function) → $\frac{C}{R} = \frac{G}{1 \pm GH}$

يعني عكس اشارة السهم

↳ characteristic equation
$1 \pm GH = 0$

Feedback Ratio → $\frac{B}{R} = \frac{GH}{1 \pm GH}$

Error Ratio → $\frac{E}{R} = \frac{1}{1 \pm GH}$

open loop transfer function → $\frac{B}{E} = GH$

Feed forward transfer function → $\frac{C}{E} = G$

* To Reduce the Block diagram to conical form
→ we look for
  - series
  - parrallel
  - Loop

* you can exchange the circle ○ and square □ by modifying the calculations ✓ Allowed

* it is not allowed to exchange a takeoff point with a circle ✗ Not Allowed

*Note:-
if the blocks are
series → ضرب
parrallel → جمع

Mason's Rule

Control Ratio:- (Closed loop transfer function)

$$\frac{C}{R} = \frac{\sum_{i=1}^{n} P_i \Delta_i}{\Delta}$$

n → # of forward paths

Δ_i → Determinant of the ith forward path

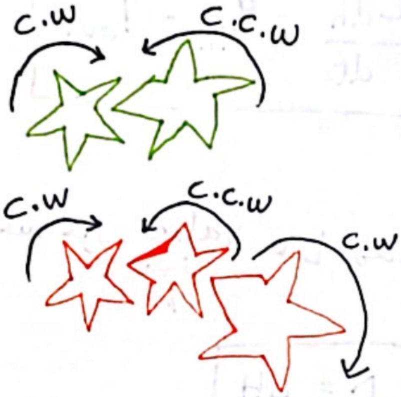
P_i → The ith forward-path gain

Δ → Determinant of the system

Gears

→ اجهزة بتعمل احتكاك وممكن توقف
ال motion زي البريك مثلاً

هدفها
- transmit motion
- change speed
- change direction



ال gear الأول
(يلي بسبب الحركة)

$$\text{*Gear Ratio} = \frac{\text{※ of teeth of input}}{\text{※ of teeth of output}} = \frac{\text{Input Torque}}{\text{Output Torque}} = \frac{\text{Output Speed}}{\text{Input speed}}$$

ال gear الذي يتحرك
نتيجة لحركة ال gear الأول

- Energy for the first gear (Driving gear) = Energy for the ~~f~~second gear (following gear)

$$N_1\,\theta_1 = N_2\,\theta_2$$

N_1: ※ of teeth for the first gear
θ_1: angular movement of the driving gear
N_2: ※ of teeth for the following gear
θ_2: angular movement of the following gear

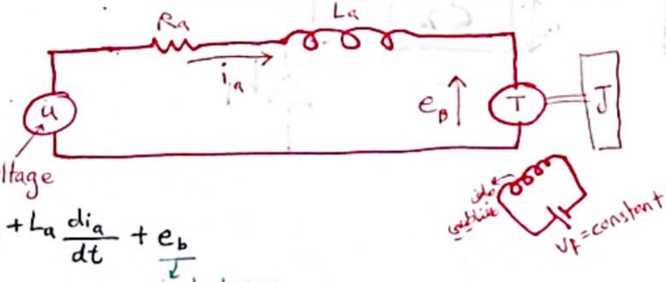
قانون ال equivilant inhertia →

$$J_{eq} = J_1 + \left(\frac{N_1}{N_2}\right)^2 J_2 + \left(\frac{N_1}{N_2}\right)^2\left(\frac{N_3}{N_4}\right)^2 J_3$$

DC Motors

المبدأ لـ هو قلب القطبية ←

→ عبارة عن Electrical and Mechanical systems مع بعض



- Torque = constant x current
  علاقة طردية ← (T, I)
- e_B = constant x ω
  علاقة طردية ← (e_B, W)

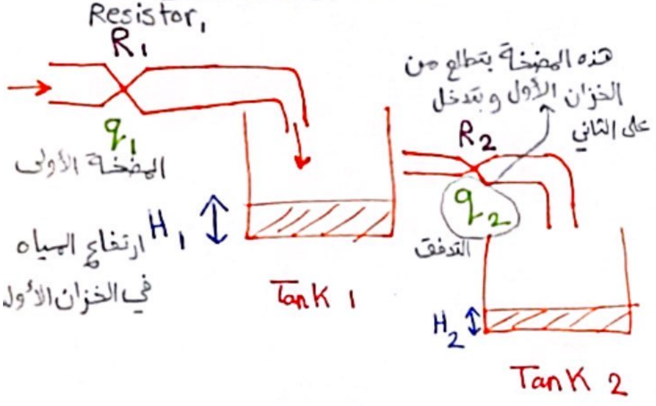
Electrical subseystem
↳ $u = R_a i_a + L_a \frac{di_a}{dt} + e_b$
e_b = constant x w (omega)

Mechanical Subsystem
↳ $J\dot{w} + B\,w$

$\dot{\theta} = w$
$\ddot{\theta} = \dot{\omega}$

Resistance of Liquid - Level Systems :-

Tanks



① سوف نكتب معادلة عند كل Tank

هنا 2 tanks
اذا 2 equations

$$\boxed{\frac{C\,dh}{dt} = q_{in} - q_{out}}$$

② عند كل valve (مضخة (q)) عنا معادلة لـ R

$$\boxed{R = \frac{\Delta H}{q}}$$

ΔH ← يلي بيدخل - يلي بيطلع

الوحدة $\frac{m}{m^3/s}$

1

$$\frac{C_1 dh_1}{dt} = q_1 - q_2$$

q_1: المضخة يلي بتدخل هي عالخزان

q_2: المضخة يلي بتطلع هي من الخزان

2

$$\frac{C_2 dh}{dt} = q_2 - 0$$

1

$$R_1 = \frac{0 - h_1}{q_1}$$

2

$$R_2 = \frac{h_1 - h_2}{q_2}$$

*Note :-

Position $\xrightarrow{\text{مقابل}}$ Voltage

Force $\xrightarrow{\text{مقابل}}$ Current Source

Damper $\xrightarrow{\text{مقابل}}$ Inductor

Mass $\xrightarrow{\text{مقابل}}$ group of capacitors

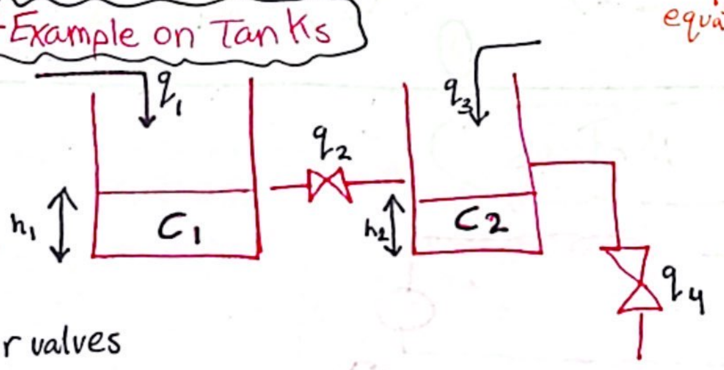
*The Transfer Function is :

$$\frac{H(s)}{Q_i(s)} = \frac{R}{(RCs+1)}$$

---

*عدد الvalves = عدد الequations
4q اذا 4

*Example on Tanks



for tank ①

$$\frac{C_1 dh}{dt} = q_1 - q_2$$

for tank ②

$$\frac{C_2 dh}{dt} = (q_2 + q_3) - q_4$$

*and now the equations for valves

$$R_1 = \frac{0 - h_1}{q_1} \qquad R_3 = \frac{0 - h_2}{q_3}$$

$$R_2 = \frac{h_1 - h_2}{q_2} \qquad R_4 = \frac{h_2 - 0}{q_4}$$

Continue → Basic Types of Mechanical Systems

Rotational

- Rotational Spring $T = K(\theta_1 - \theta_2)$
  (K)
- Rotational Damper $T = C(\dot{\theta}_1 - \dot{\theta}_2)$
  (C)
  نقاوم من جهة θ_1
- Moment of inhertia $T = J\ddot{\theta}$
  (J)

هنا ال Torque تقابل ال Force

θ تقابل ال Position

J تقابل ال Mass

*power = Force x Velocit
F $\dot{x}$

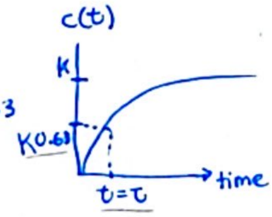
Transient and Steady State Response Analysis

Time Response consists of two parts $\begin{cases}\text{Transient Response} \\ \text{steady State Response}\end{cases}$

*First Order System :-

Transfer Function $\dfrac{C(s)}{R(s)} = \dfrac{1}{\tau s+1}$

after taking inverse Laplace Transform $\boxed{c(t) = 1 - e^{-\frac{t}{\tau}}}$



*if $t=\tau=RC$ $\quad c(\tau) = 1 - .37 = 0.63$

$T_r = 2.2\tau$

$T_s = 4\tau$

← كل ما كان مقدار
الجذر أقل كلما كان أبطأ

$\tau = \frac{1}{a}$ $\quad \tau = RC$

* Second Order System :-

[illegible] complex

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

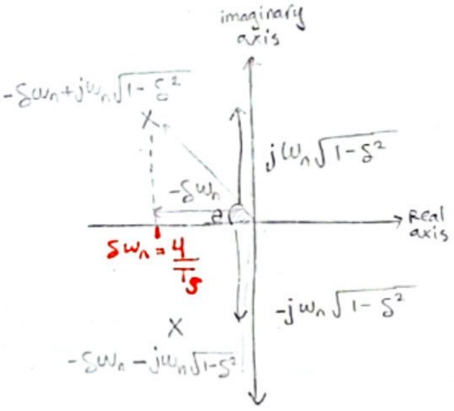
undamped natural frequency $\omega_n = \sqrt{b}$

$\zeta = \dfrac{a}{2\sqrt{b}}$ (zeta) damping Ratio

$$\boxed{G(s) = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}}$$

Poles $\begin{cases} \to -\omega_n\zeta + \omega_n\sqrt{\zeta^2-1} \\ \to -\omega_n\zeta - \omega_n\sqrt{\zeta^2-1}\end{cases}$

poles are complex if $(\zeta<1)$



For transient Response :-

$T_r = \dfrac{\pi - \theta}{(\omega_n\sqrt{1-\zeta^2})}$ θ in Radians; $(\omega_n\sqrt{1-\zeta^2})$ → complex part

$T_p = \dfrac{\pi}{\omega_n\sqrt{1-\zeta^2}}$

$\%MP$
$\%OS$ $= e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \times 100\,\% \xrightarrow[\zeta \text{ will be}]{\text{if OS\% is given}} \zeta = \dfrac{-\ln(\%MP/100)}{\sqrt{\pi^2+\ln^2(\%MP/100)}}$

$T_s = \dfrac{4}{\zeta\omega_n}$

$\theta = \cos^{-1}(\zeta)$

According to (ζ)

- overdamped → 2 Real poles → $\zeta>1$
- underdamped → 2 complex conjugate poles → [illegible]
  ↳ $0<\zeta<1$
- sinusoidal → undamped → 2 imaginary poles → $\zeta=0$
- critically damped → 2 real but equal poles → $\zeta=1$

Steady State Error

e_{ss}

e_{ss}

step $\frac{1}{s} \rightarrow \frac{1}{1+K_p}$

Ramp $\frac{1}{s^2} \rightarrow \frac{1}{K_v}$

parabola $\frac{1}{s^3} \rightarrow \frac{1}{K_a}$

steady state error for

System Type	Step	Ramp	Parabolic
0	$\frac{1}{1+K_p}$	∞ $K_v=0$	∞
1	0	$\frac{1}{K_v}$ $K_v=\text{finite}$	∞
2	0	0 $K_v=\infty$	$\frac{1}{K_a}$
3			0

$$\text{Error} = R(s) - C(s)$$

$$e_{ss} = \lim_{t \to \infty} e(t) \Rightarrow \lim_{s \to 0} s\,E(s)$$

$$\text{Error}\,(E(s)) = \frac{1}{1 + G(s) \times \frac{1}{H}} \times R(s)$$

$$\lim_{s \to 0} s \times \frac{1}{1+G} \times R$$

$$K_p = \lim_{s \to 0} G(s)$$

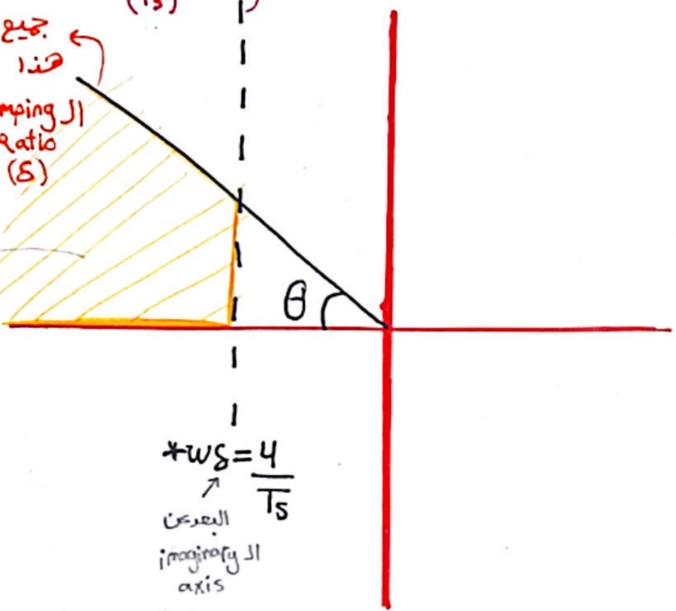
static velocity error constant ← $K_v = \lim_{s \to 0} s \times G(s)$

$$K_a = \lim_{s \to 0} s^2 \times G(s)$$

Root Locus

أي نقطة على هذا العمود تمثل constant settling Time (T_s)

جميع النقاط على هذا الخط لها نفس الـ damping Ratio (δ) والـ Percent overshoot $(OS\%)$



أي نقطة ضمن هذا الـ Region تحقق الشرط

$$*w\delta = \frac{4}{T_s}$$

البعد عن الـ imaginary axis

*كلما تحرك العمود لليسار قل الـ Ts

*جذور المقام هي التي تحدد
- settling time
- Transient responge

*المؤثر الحقيقي
- لما يكون عندي جذر واحد (one zero)
- أو جذرين complex (Dominant Poles)

The closed loop Transfer Function is

$$\frac{C}{R} = \frac{G}{\underbrace{1+GH}_{\text{characteristic equation}}}$$

$$1 + GH = 0$$

$$GH = -1$$

it is a complex quantity so it can split to
- angle
- magnitude

- **the angle :-**

*لو طلعت الزاوية 180° أو مكرراتها إذاً هي نقطة بقدر اوصلها عن طريق تغيير قيمة (K)

$$\angle GH = \pm 180^\circ (2K+1)$$

where K=1, 2, 3,..

- **the magnitude :-**

$$|GH| = 1$$

PID

* First Method

$$K_o = \frac{y_\infty - y_o}{u_\infty - u_o}$$

$$\tau_o = t_1 - t_o$$

$$V_o = t_2 - t_1$$

Type of controller	K_p	T_r	T_d
P	$\frac{V_o}{K_o \tau_o}$		
PI	$\frac{0.9\, V_o}{K_o \tau_o}$	$3\tau_o$	
PID	$\frac{1.2\, V_o}{K_o \tau_o}$	$2\tau_o$	$0.5\tau_o$

* Second Method

$S = j\omega$

$P_{cr} = \frac{2\pi}{\omega}$

Type of controller	K_p	T_i	T_d
P	$0.5 K_{cr}$	∞	0
PI	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID	$0.6 K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

بعدين اذا طلب السؤال ~~then~~ transfer function

$$\rightarrow G = K_p \left(1 + \frac{1}{T_i S} + T_d S\right)$$

Aurdoino

Snippet from code to switch LED < ON / OFF by listening to a button

```
const int BUTTON = 12 ;
const int LED    = 13 ;

void setup ()
{
  PinMode (BUTTON, INPUT);
  PinMode (LED , OUTPUT);
}
 void loop ()
{ val = digitalRead (BUTTON);
    if (val == HIGH)
  {  digitalWrite (LED, HIGH); }
     else
   { digitalWrite (LED, LOW); }

 }
```

Blink LED

```
const int LED = 13 ;

 void  setup ()
 {
   PinMode (LED, OUTPUT);
 }

  Void loop ()
 {
  digitalWrite (LED, HIGH);
   delay (1000);
  digitalWrite (LED, LOW);
   delay (1000);
 }
```

TMP 36

```
 const int temperaturePin = 0 ;
void setup()
{ Serial.begin (9600); }
 void loop ()
{
 float voltage, degreesC, degreesF ;
 voltage = getVoltage (tempraturePin);
degrees C = (voltage - 0.5)* 100.0 ;
  serial.print (" voltage; ");
  Serial.print (voltage );
  Serial.print (" deg C:");
  Serial.println (degrees C);
   delay (1000);
}
 float get Voltage (int pin)
 { return(analogRead (pin) * 0.004882814;}
                           ↳ 5/1024
```

Thermistor

```
  const int temperaturePin = 0 ;
 void setup ()
{ Serial.begin (9600); }
 void loop ()
{ int temprature = getTemp () ;
 Serial.print ("Temperature Value : ");
 Serial. print  ( temperature);
 Serial.print ln (" *C ");
   delay (1000);
}
  double getTemp ()
 {
  analogRead (temperaturePin);
  long Resistance;
  double Temp ;
 Resistance = ((10240000/RawADC)- 10000);
Temp = 1/((0.001129148 + (0.000234125 *ln(Resistance)) +
      (.0000000876741 * ln(Resistance) * ln(Resistance))
```

Morse Code

o — — —

s • • •

```
void dot ()
{ blink (200, 200); }


void dash ()
{ blink (600,200); }

void letterspace ()
{ delay (400); }

void wordspace ()
{ delay (800); }


void morse_s()
{ dot(); dot;(); dot(); letterspace (); }


void morse_o()

{ dash(); dash(); dash; letterspace(); }
```

Matlab

$'' \rightarrow s^2$

$' \rightarrow s$

* $y'' + 6y' + 5y = 4u' + 3u$

$s^2Y(s) + 6s\,Y(s) + 5\,Y(s) = 4s\,u(s) + 3u(s)$

Transfer $\rightarrow$ function $G(s) = \frac{Y(s)}{u(s)} = \frac{4s+3}{s^2+6s+5}$

$\xrightarrow[\text{function}]{\text{by zpk}}$ $\frac{4s+3}{s^2+6s+5} = \frac{4(s+0.75)}{(s+1)(s+5)}$

```
>sys1 = zpk(-0.75,[-1 -5],4);
```

On Matlab :-

```
>> num = [4 3] ;
>> den = [1 6 5] ;
>> sys = tf (num, den) ;
```

or

easiest way to write a transfer function:-

```
>> s = tf {'s'}
>> g = (4*s +3)/(s^2+6*s+5)
```

* اذا طلب السؤال step

```
>>step (feedback(g,-1))
```

* اذا طلب السؤال Root locus

```
>> rlocus (g)
```

* Question :-

Plot the root locus of the following system

$$G(s) = \frac{K(s+8)}{S(s+2)(s^2+8s+32)}$$

Method ①

```
>>num = [1 8];
>> den = conv(conv([1 0],[1 2]), [1 8 32]);
>>sys = (num,den);
>> rlocus (sys)
```

Method ② in one step الحل

```
rlocus(tf([1 8],con(conv([1 0],[1 2]),
    [1 8 32])))
```

State Space Models

$$y'' + 6y' + 5y = 4u' + 3u$$

$$\frac{Y(s)}{u(s)} = \frac{4s+3}{s^2+6s+5} = \frac{1}{s^2+6s+5} * (4s+3)$$

$$\boxed{\frac{X}{R} = \frac{1}{s^2+6s+5}} \qquad \boxed{\frac{Y}{X} = 4s+3}$$

هنا بجعل R موضوع القانون

هنا بجعل Y موضوع القانون

$$R = X'' + 6X' + 5X \qquad Y = 4X' + 3X$$

$$X_1' = X_2$$

$$X_2' = (-5X_1 - 6X_2) + R$$

In Matrix Form:-

$$\begin{bmatrix} X_1' \\ X_2' \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -5 & -6 \end{bmatrix} * \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} R$$

$$Y = \begin{bmatrix} 3 & 4 \end{bmatrix} * \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} R$$

Solving Differential Equation using MATLAB

Bacteria Population

birth rate $= bx$

death rate $= px^2$

total range of change of bacteria population

$$\dot{x} = bx - px^2$$

$b = 1/\text{hour}$

$p = 0.5/\text{hour}$

time $[0 \quad 1]$

assuming that initially → $x_0 = 100$ bacteria

Solution:-

```
function dx = bacteriadiff(t,x)

b = 1 ;
p = 0.5 ;
dx = b*x-p*x^2 ;

tspan [0 1];
Xo = 100  ;
[t,y] = ode45(@bacteriadiff,tspan,xo);
plot (t,y)
```

Passing Parameters to the model

1st order diffrential equation:

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

whee $a = -\frac{1}{T}$

we want a, b as parameters to make them easy to change valu

$b = 1$

$X_0 = 1$

$T = 5$

Solution:-

```
function dx = mysimplediff (t,x,param)
a = param(1) ;
b = param(2);

dx = a*x + b ;

tspan = [0 25];
Xo = 1;
a = -1/5;
b = 1 ;
param = [a b];
[t,Y] = ode45(@mysimplediff, tspan, xo,[],param)
plot (t,Y)
```

Diffrential Equation

* Use ode23 and plot the results in the interval $[t_0, t_f]$

$$\left[\dot{w} + (1.2 + \sin 10t)\, w = 0\right] \rightarrow \dot{w} = -(1.2 + \sin 10t) w$$

where $t_0 = 0$

$t_f = 5$

$w(t_0) = 1$

Solution:-

```
function dw = diff_task3(t,w)
dw  = -(1.2+sin(10*t))*w ;

tspan = [0 5];

wo = 1;

[t,w] = ode23(@diff_task3,tspan,wo);

plot (t,w)
```

2nd ORder differential

use ode23

$$(1 + t^2)\ddot{w} + 2t\dot{w} + 3w = 2$$

$$\rightarrow \ddot{w} = \frac{2 - 2t\dot{w} - 3w}{1 + t^2}$$

$t_0 = 0$

$t_f = 5$

$w(t_0) = 0$

$\dot{w}(t_0) = 1$

$$\ddot{w} = \dot{x}_2 = \frac{2 - 2t x_2 - 3x_1}{1 + t^2}$$

لازم احولة لدرجة أولى

$\boxed{w = x_1}$

$\boxed{\dot{w} = x_2}$

$\boxed{\ddot{w} = \dot{x}_2}$

$\dot{x}_1 = x$

Solution:-

```
function dx = diff_secondorder(t,x)
[m,n] = size(x);
 dx = zeros(m,n)*
 dx(1) = x(2) ;
 dx(2) = (2-2*t*x(2) -3*x(1))/(1+t^2)

tspan [0 5];
X0 = [0 ; 1]

[t,x] = ode23 (@diff_secondorder,tspan,xo);
 plot (t,x)
 legend ('x1','x2')
```

First order

$$\frac{dy}{dt} = ay$$

```
syms y(t) a
eqn =diff(y,t) == a*y ;
 S = dsolve(eqn
```

$S = C_1 e^{at}$

Second order

$$\frac{d^2y}{dt^2} = ay$$

```
syms y(t) a
eqn = diff(y,t,2)= = a*y ;
 S = dsolve(eqn)
```

$S = C_1 e^{-\sqrt{at}} + C_2 e^{\sqrt{at}}$

Second Order with initial conditions

$$\frac{d^2y}{dt^2} = a^2 y$$

$y(0)=b$

$\boxed{y'(0)=1}$

```
syms y(t) a b
eqn = diff(y,t,2) = = a^2 *y;
Dy = diff(y,t) ;
cond = [ y(0)= = b , Dy(0) = =1 ];
 S = dsolve(eqn,cond)
```

System of differential equations

$$\frac{dy}{dt} = z \qquad \frac{dz}{dt} = -y$$

without assignment

```
syms y(t) z(t)
eqns = [diff(y,t) = = z ,diff(z,t)= =-y]
 S = dsolve(eqns)
```

with assignment

```
syms y(t) z(t)
eqns = [diff(y,t)= =z ,diff(z,t)= =-y]
[ysol(t),zsol(t)] =dsolve(eqns)
```

another question

```
syms y(t)
eqn = diff(y)= = y+exp(-y)
sol = dsolve(eqn)
```