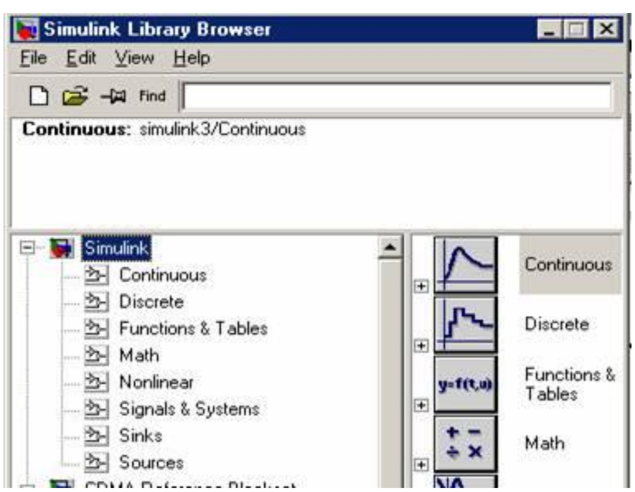
SIMULINK – MATLAB

CONTENIDO

1. **ELEMENTOS BÁSICOS**
2. **EL MOTOR DC**
3. **SUBSISTEMAS**
4. **ECUACIONES DIFERENCIALES**
5. **SIMULACIÓN DE SISTEMAS**

INTRODUCCIÓN

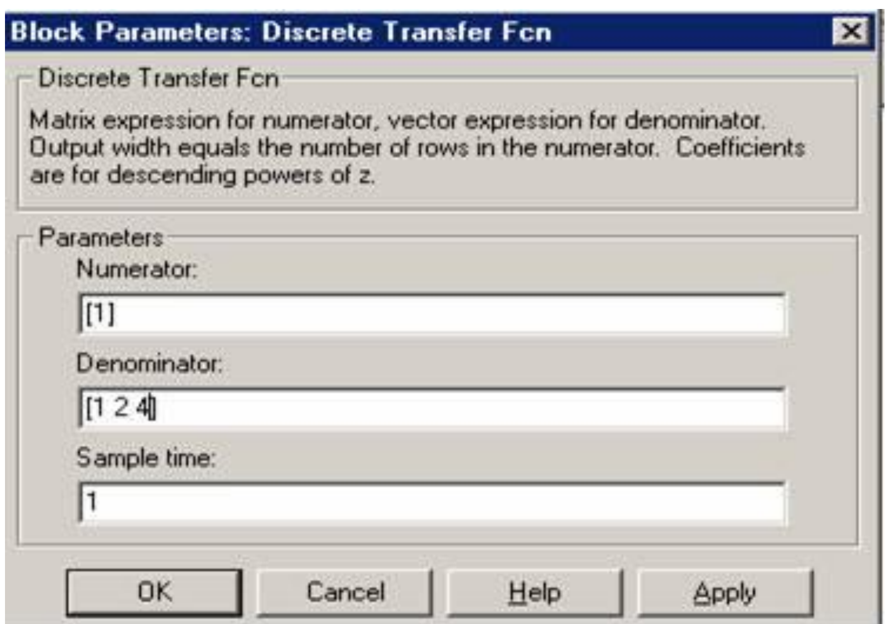
Simulink es una extensión de Matlab utilizado en el modelamiento y simulación de sistemas. Para arrancar Simulink se puede hacer desde el prompt de Matlab digitando el comando >>Simulink o utilizando el icono . Se abre la ventana Simulink Library Browser como se indica abajo y se puede diagramar un nuevo modelo activando el botón New Model , o sea el icono o de File □ New□Model



Un modelo es un conjunto de bloques que representa un sistema y como archivo tiene extensión *.mdl

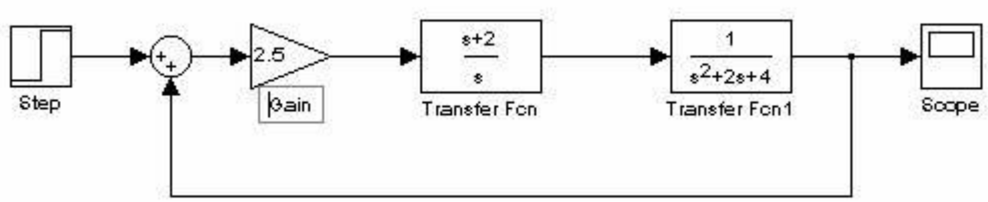
1. ELEMENTOS BÁSICOS

Los elementos básicos son líneas y bloques. Los bloques están agrupados en: Sources, Links, Discrete, Continuos, Math, etc., tal como aparecen en la ventana anterior. Cada bloque tiene entradas y salida para realizar su interconexión. Por ejemplo, haga clic en Discrete y luego clic en Discrete Transfer Fcn y arrastre el bloque a la ventana en blanco. Si quiere modificar la función de transferencia del bloque haga doble clic en él y digite los coeficientes del numerador y denominador en la nueva ventana que aparece. Para la función $1/(z^2 +2z +4)$ con tiempo de muestreo de 1 seg, quedaría:

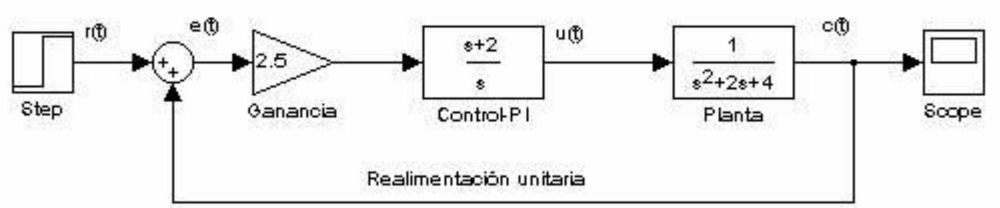


Para realizar el diagrama en bloques de un sistema se hace lo siguiente:

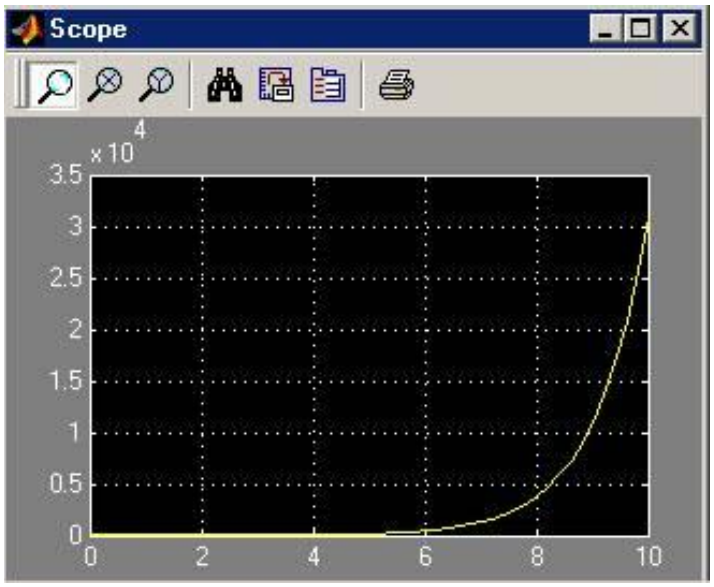
Lo primero es arrastrar los bloques a la página en blanco de forma que, Step es la función paso o escalón que se obtiene de Sources, Scope es el osciloscopio que se obtiene de Sinks, Transfer Fcn se obtiene de Continuos, Sum y Gain se obtienen de Math. Modifique los bloques dando doble clic sobre cada uno de ellos para cambiar sus parámetros o valores e interconéctelos.



Lo segundo es cambiar los nombres a los bloques y asignar las variables o señales haciendo doble clic en el lugar en que se van a colocar y salvar el modelo especificándole un nombre, por ejemplo ejem1.mdl



Por último se debe simular el sistema. Para ello se configura la señal de entrada, en este caso la función paso. Dar doble clic y asignar los siguientes parámetros: Step time=0, Inicial value=0, Final value=1, Sample time=0. Para simular el sistema de control se escoge del menú Simulation □ Start o el icono .y luego se hace doble clic en Scope para ver su respuesta o salida del sistema. Para observar además la entrada se puede colocar otro Scope a la salida de Step y se puede probar para varios pasos variando su amplitud, tiempo de inicio y tiempo de iniciación del paso. Para observar mejor la respuesta se usa el botón Autoscale (binoculares) de la ventana del Scope. Si quiere observar mejor la respuesta o parte de ella se pueden cambiar los parámetros de simulación, Simulation→ Simulation parameters. Por ejemplo cambiar el *Start time* y el *Stop time* y correr nuevamente la simulación.

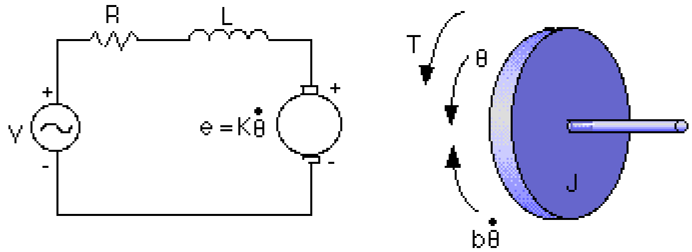


2. EJEMPLO: MODELAR UN MOTOR DC

Un actuador común en sistemas de control es el motor DC. Provee directamente movimiento rotatorio y acoplado con poleas o correas puede proveer movimiento transnacional.

2.1 ECUACIONES DINÁMICAS

El circuito eléctrico de la armadura y el diagrama de cuerpo libre del rotor es mostrado en la figura con sus ecuaciones dinámicas.



(1) Leyes de Newton

$$J\frac{d^2\theta}{dt^2} = T - b\frac{d\theta}{dt} \Rightarrow \frac{d^2\theta}{dt^2} = \frac{1}{J}\left(k_t i - b\frac{d\theta}{dt}\right), \quad \text{debido a que}, T = k_t i$$

(2) Leyes de Kirchhoffs

$$L\frac{di}{dt} = -Ri + V - e \Rightarrow \frac{di}{dt} = \frac{1}{L}\left(-Ri + V - k_e\frac{d\theta}{dt}\right), \quad \text{debido a que}, e = k_e\frac{d\theta}{dt}$$

Los parámetros físicos tienen los siguiente valores :

Momento de inercia del rotor : $J = 0.01 kg.m^2/sg^2$

Rata de amortiguamiento del sistema mecánico: b = 0.1 N.m.sg

Constante de la fuerza electromotriz: Ke = Kt = 0.01 Nm/Amp

Resistencia eléctrica: R = 1 ohm

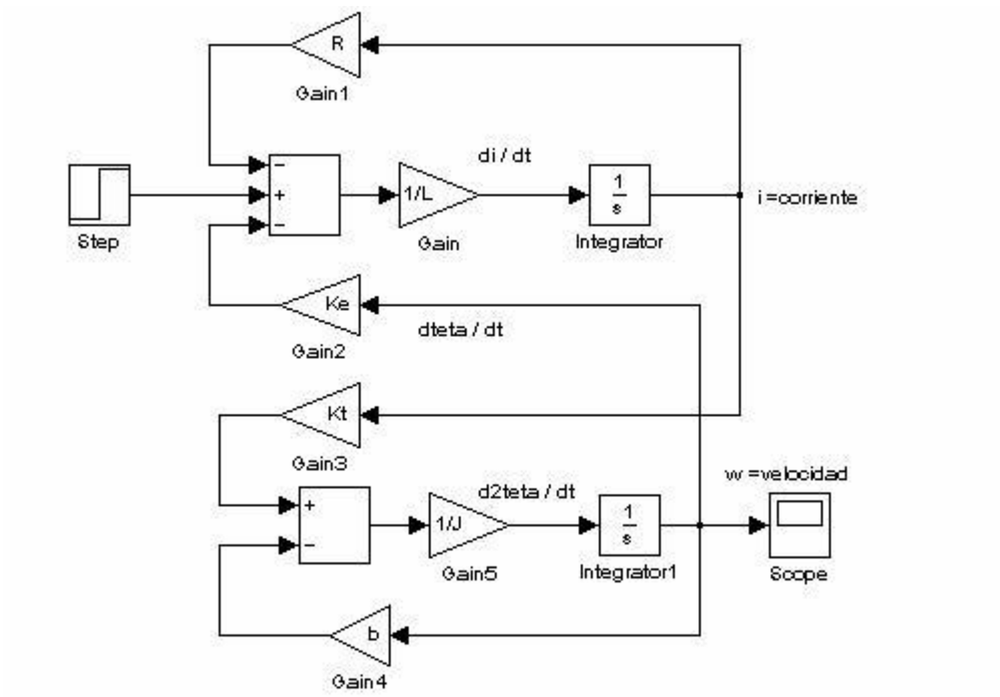
Inductancia eléctrica: L =0.5H

Fuente de voltaje de entrada: V

Posición angular: θ

Se asume que el rotor y el eje son rígidos

2.2 MODELADO DEL MOTOR EN VELOCIDAD



2.3 EXTRAER MODELO LINEAL

Para obtener la función de transferencia del motor primero se trasladan los parámetros del motor al modelo creando un archivo en Matlab (*.m) de la siguiente forma:

% VALORES DE LOS PARÁMETROS DEL MOTOR

J = 0.01;

b = 0.1;

Ke = 0.01;

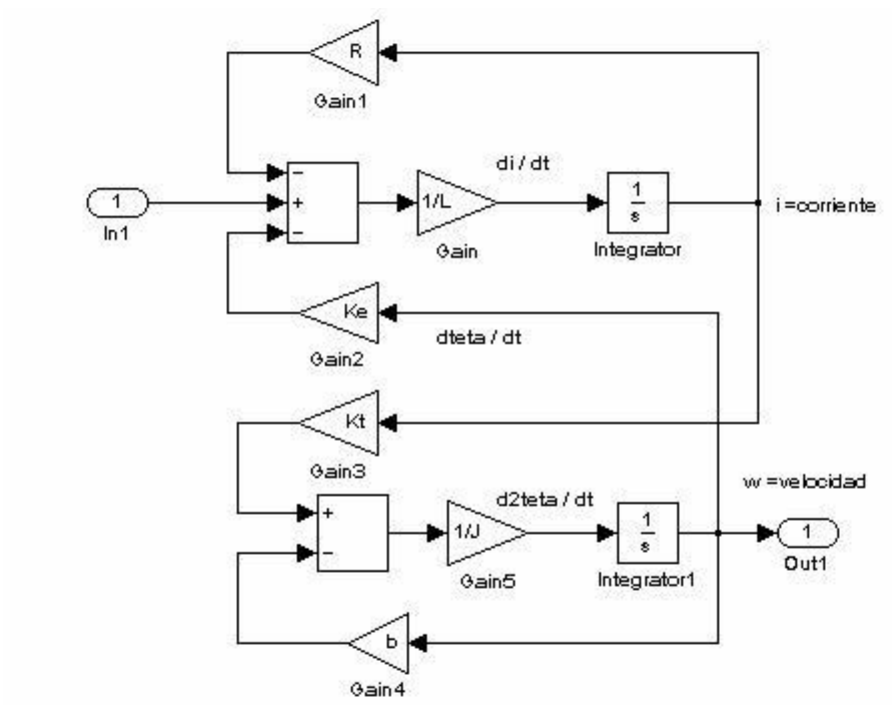
Kt = 0.01;

R = 1;

L = 0.5;

Se ejecuta este archivo y se simula el modelo para una entrada de paso unitario de valor V = 0.01, con los siguientes parámetros de simulación: Stop time = 3 sg. Arranque la simulación y observe la salida (velocidad del motor).

Como segundo paso se debe obtener el modelo lineal de Matlab del motor. Para esto, borre el bloque Scope y cámbielo por Out obtenido de la librería de Signals&Systems. Haga lo mismo para Step cambiándolo por In de esta misma librería. Los bloques In y Out definen la entrada y salida del sistema que le gustaría extraer. Salve este modelo. El sistema quedará así:



Como tercero y último paso, después de desarrollado el modelo y salvarlo por ejemplo con el nombre MotorDcVel.mdl se ejecutan los siguientes comandos:

% OBTENER EL MODELO LINEAL DEL SISTEMA

[num, den] = linmod('MotorDcVel')

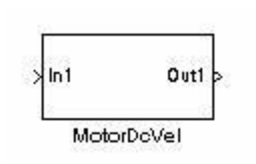
Gps = tf(num, den)

La respuesta es :

$$Gp(s) = \frac{2}{s^2 + 12s + 20.02}$$

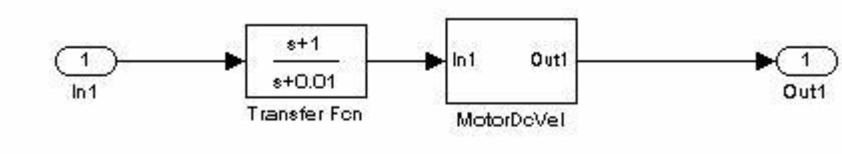
3. SUBSISTEMAS

Abra una nueva ventana y arrastre de la librería Signals&Systems el bloque SubSystem , haga doble clic en este bloque, abra el modelo MotorDcVel.mdl (el que tiene In y Out como terminales) cópielo y péguelo en la nueva ventana de subsistema anterior. Cierre ventanas y aparece una nueva con el bloque con los terminales del subsistema creado. Déle el nombre MotorDcVel. Si a este bloque de subsistema se le da doble clic aparece el modelo completo diseñado anteriormente. Otra forma es señalar los bloques de interés, ir a menú Edit --> create Subsytem



3.1 SISTEMA EN LAZO ABIERTO

Al subsistema creado que constituye la planta de un sistema de control se le va a adicionar un controlador y obtendremos la función de transferencia en lazo abierto y lazo cerrado.



% CONTROL DE UN MOTOR DC

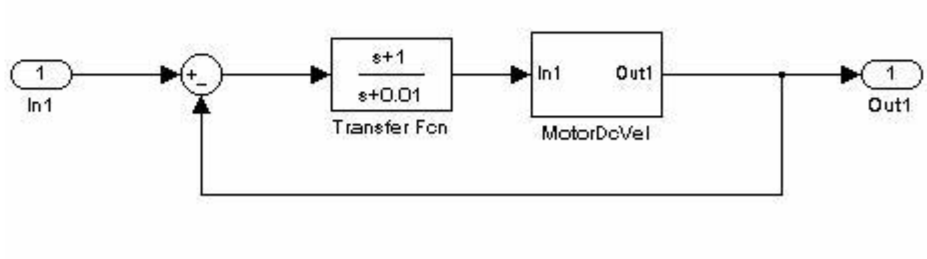
[num, den]=linmod('ControlMotor')

Glazo_abierto = tf(num, den)

Respuesta:

$$Glazo\_abierto = \frac{2s+2}{s^3+12.01s^2+20.14s+0.2002}$$

3.2 SISTEMA EN LAZO CERRADO



% CONTROL DE UN MOTOR DC

[num, den]=linmod('ControlMotor')

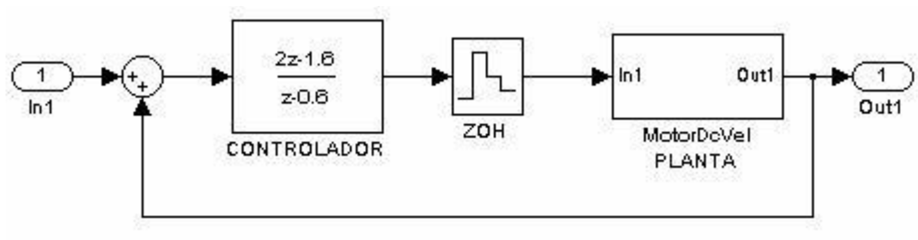
Glazo_cerrado= tf(num, den)

Respuesta:

$$Glazo\_cerrado = \frac{2s+2}{s^3+12.01s^2+22.14s+2.2}$$

3.3 SISTEMA DISCRETO

DIAGRAMA EN SIMULINK



PROGRAMA MATLAB

```
% SISTEMA DISCRETO DISCRETO
T=0.1;
[num,den]=dlinmod('MotorDigital',T)
Glazo_cerradoz=tf(num,den,T)
```

Respuesta:

$$Glazo\_cerrado(z) = \frac{0.01371z^2 - 0.001764z - 0.007364}{z^3 - 1.8z^2 + 1.015z - 0.1734}$$

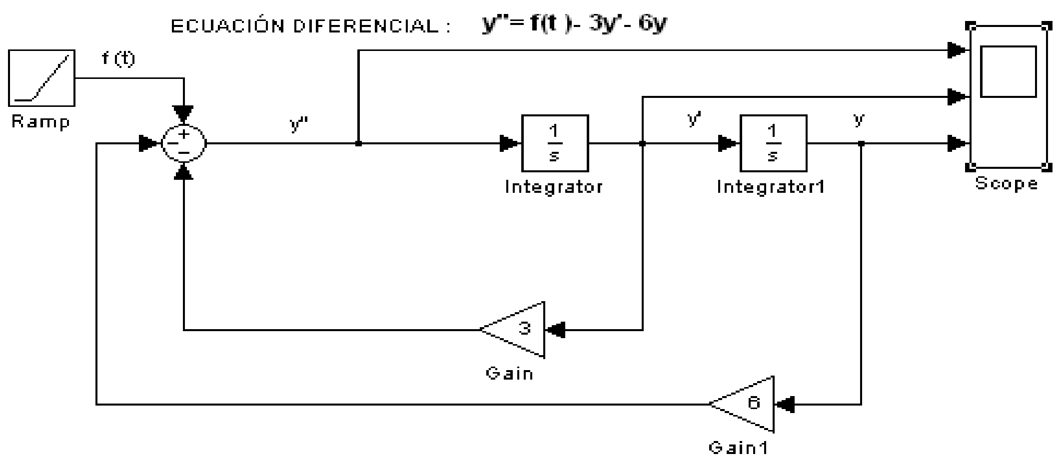
4. SOLUCIÓN DE ECUACIONES DIFERENCIALES

Ejemplo:

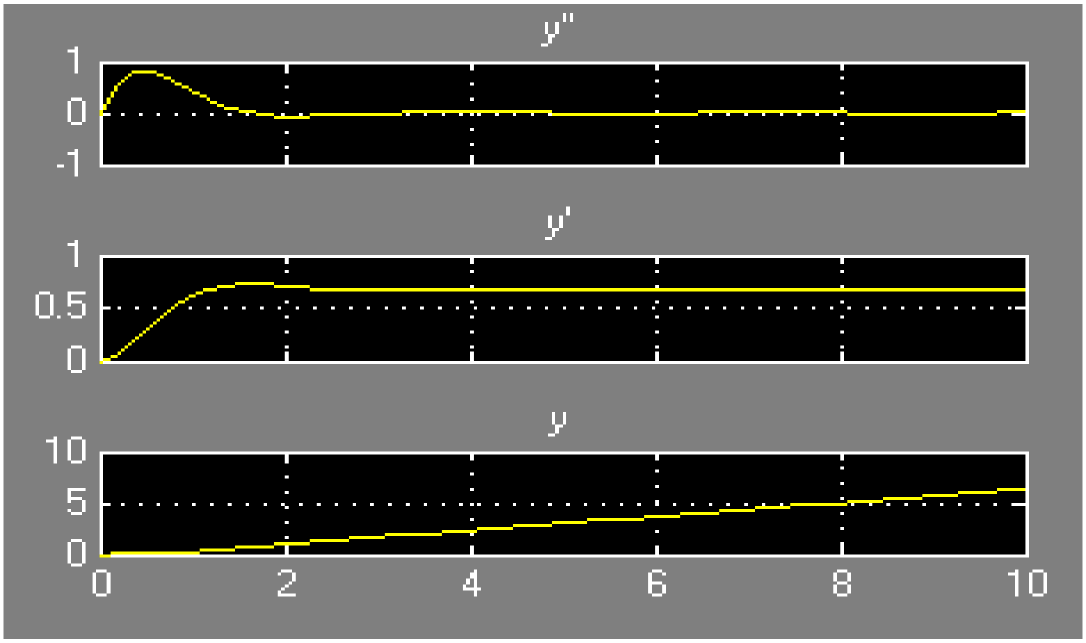
Resolver la siguiente ecuación diferencial:

$$\frac{d^2y}{dt^2}+3\frac{dy}{dt}+6y=4t \Rightarrow y''+3y'+6y=4t \Rightarrow y''=4t-3y'-6y$$

Diagrama Simulink:

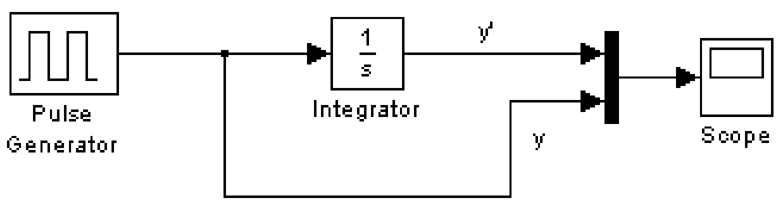


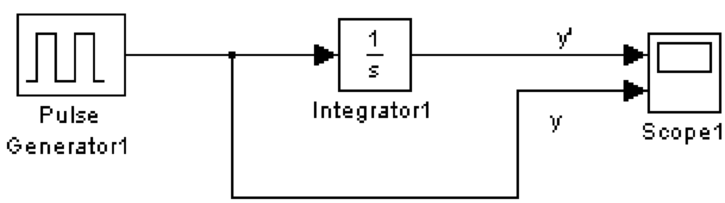
Respuestas:

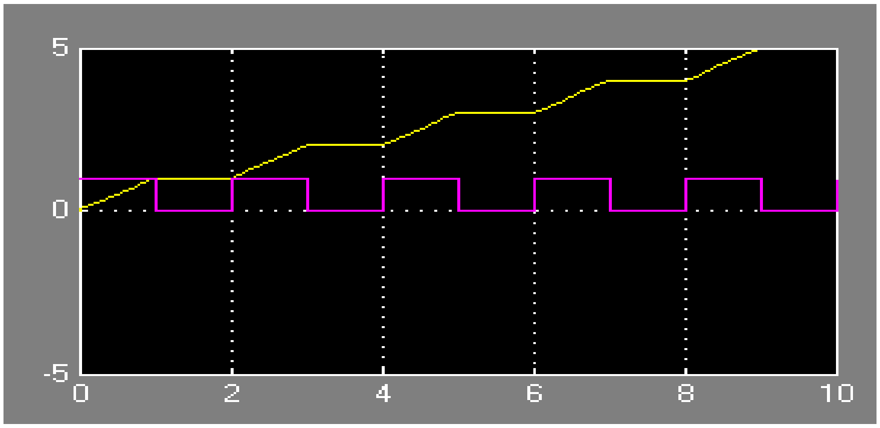


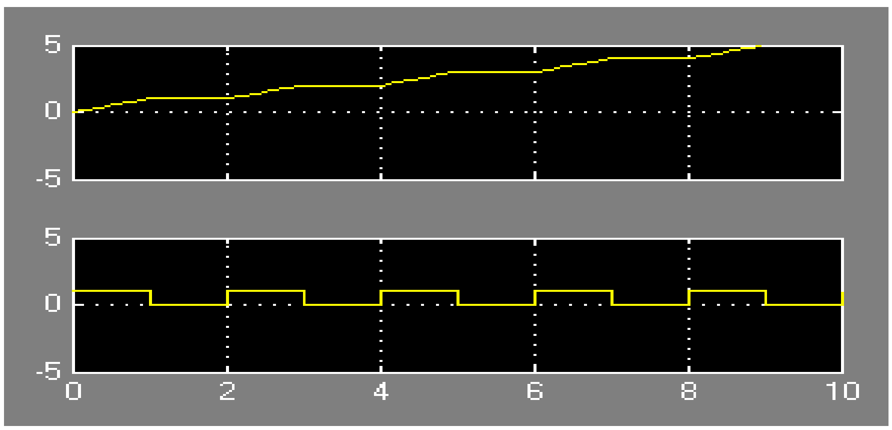
Ejemplo:

Comprobar la integración por Simulink.







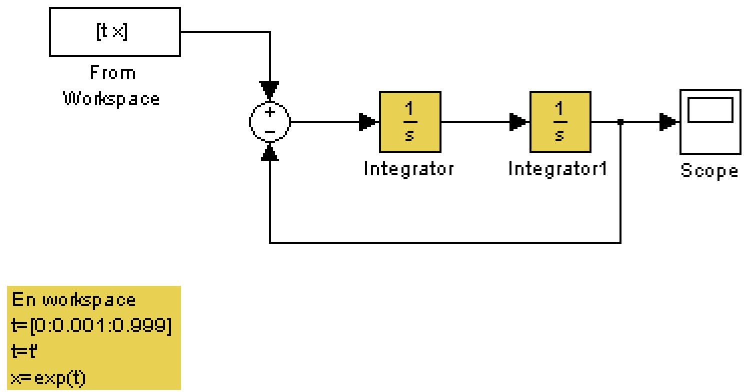


5. SIMULACIÓN DE SISTEMAS

5.1 INTERCAMBIO DE MATLAB A SIMULINK

Para utilizar señale de Matlab a Simulink de la librerís *Sources* se utiliza el bloque *From Workspace*.

Ejemplo: Resolver la ecuación $y'' + y = e^{t}$, $y'(0) = 0$, $y(0) = 3$



El vector [t x] se ejecuta en Matlab en el workspace de la siguiente forma:

```
>> t = 0:0.001:0.999;
>> t = t';
>> x = exp(t)
```

Al ejecutarse Simulink toma los datos entregados por Matlab. No olvidar colocar condición inicial y(0) = 3 en el *integrador.*

5.2 INTERCAMBIO DE SIMULINK A MATLAB

Para enviar datos de Simulink a Matlab se utiliza de la librería Sinks el bloque To Workspace.

Ejemplo:

Resolver la ecuación: $f(t) = Mx'' + Bx' + Kx$, M=1, B=1, K= 10, F(t) = 5

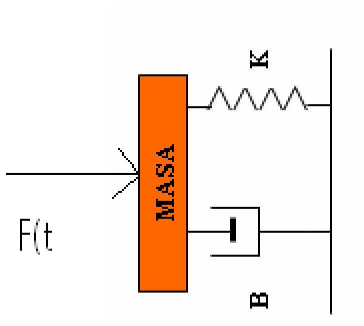
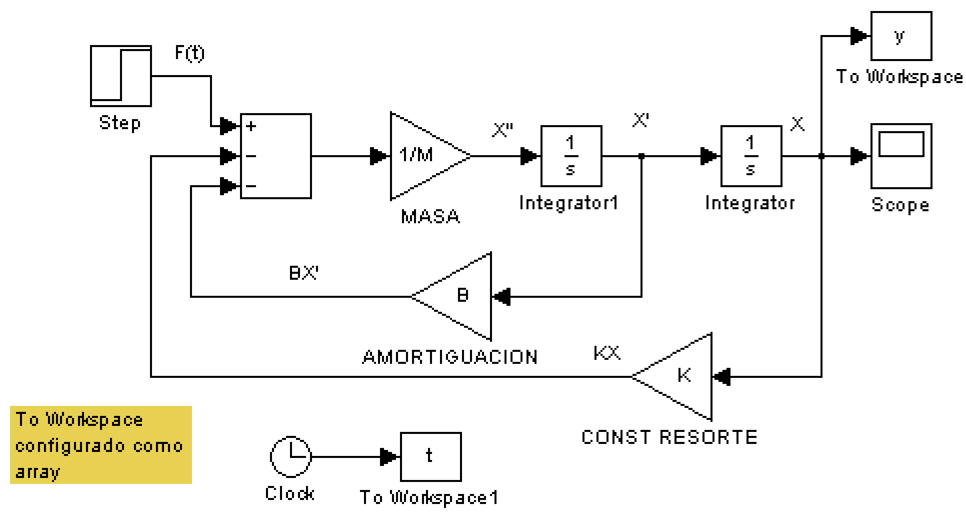


Diagrama Simulink:

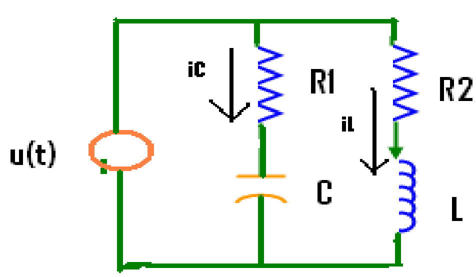


En Matlab:

>> plot(t,y)

5.3 EJERCICIOS

Ejercicio1:



Si la entrada es una señal senoidal, encontrar las salidas referidas a *vC y iL.*

Ejercicio2:

Para el siguiente problema hallar la variación de h si el caudal normal Q es de 10 lit/min y en t=5 seg se aplica una perturbación de 2 lit/min. El valor de K=10, A= 2 m^2.

$$A\frac{dh}{dt} = q(t) - K\sqrt{h}$$

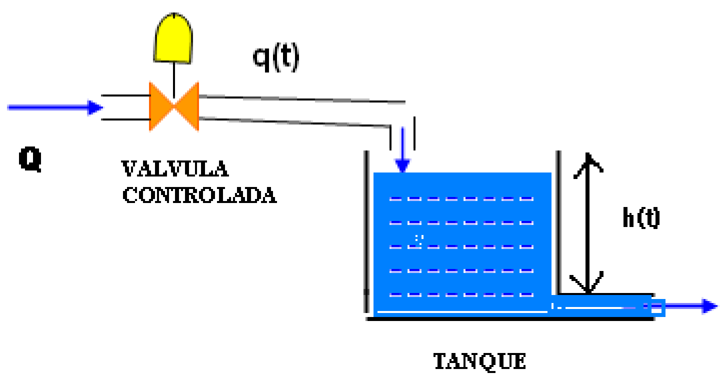
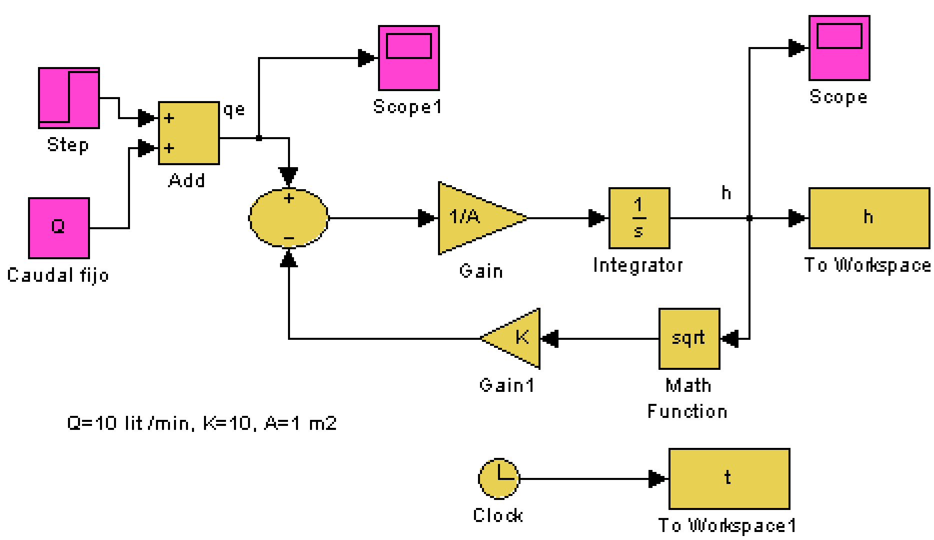
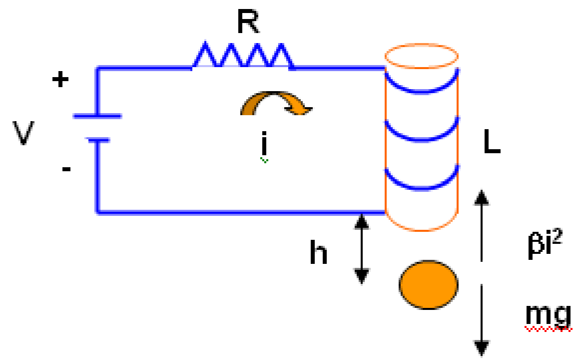


Diagrama Simulink:



EJERCICIO3: LA BOLA MAGNÉTICA



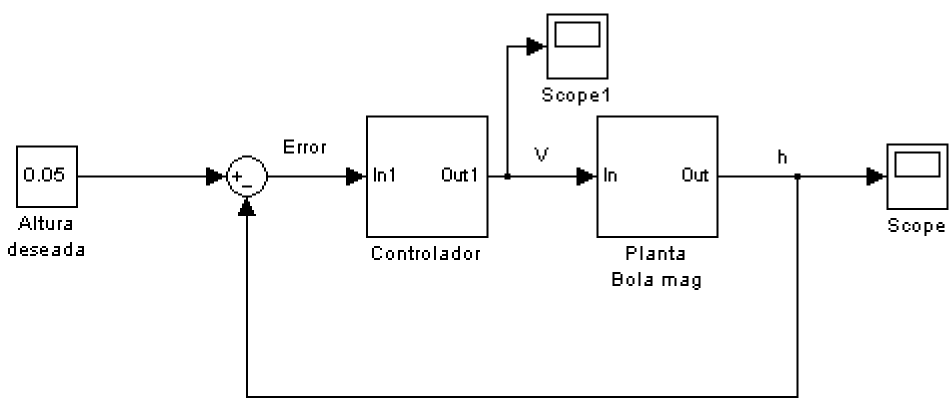
Ecuaciones:

(1) $m\frac{d^2h}{dt^2} = mg - \frac{\beta i^2}{h}$ (2) $\mathrm{L}\frac{\mathrm{di}}{\mathrm{dt}} = V - iR$

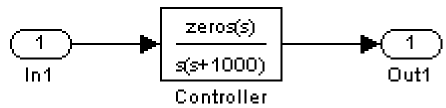
Valores:

m=0.1 Kg; g=9.81; R=2 Ohm; L=0.02 H; β=0.001

Diagrama simulink:



Controlador:

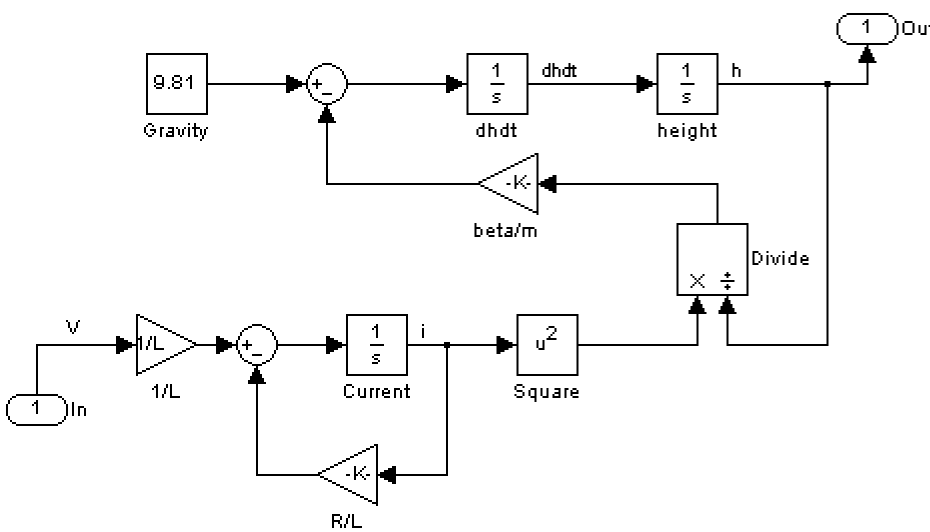


zeros=[-11.5+7.9i, -11.5-7.9i]

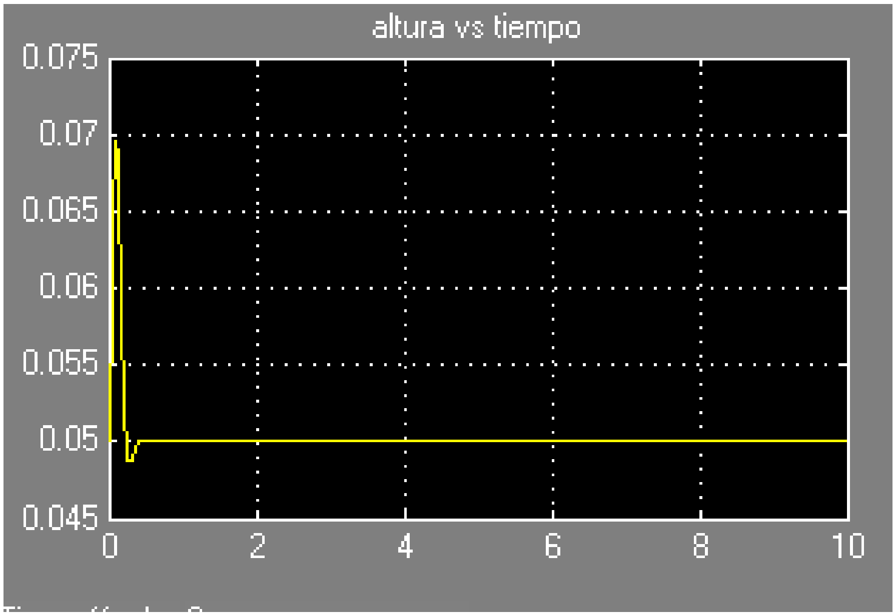
polos=[0 -1000]

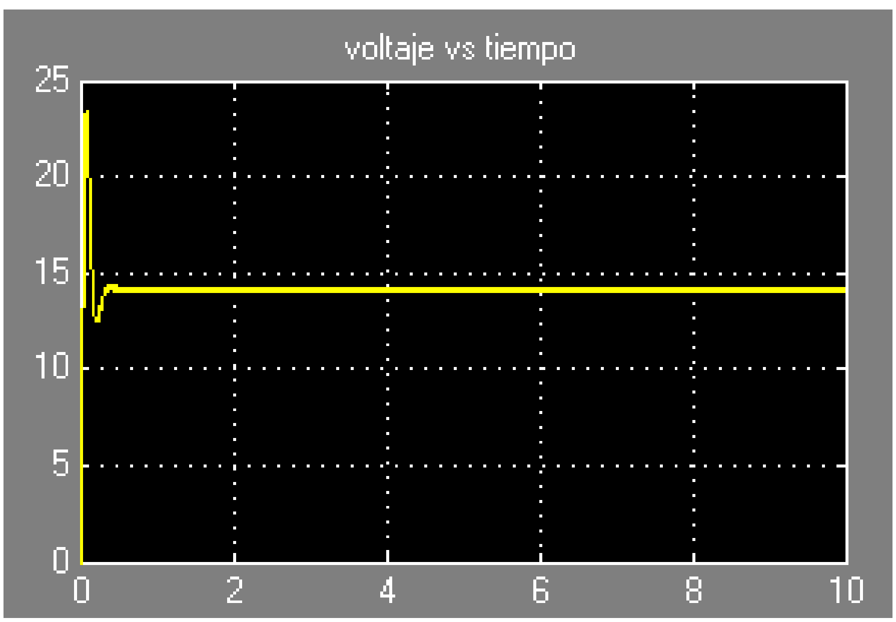
ganancia=-3.3057e+004

Planta:

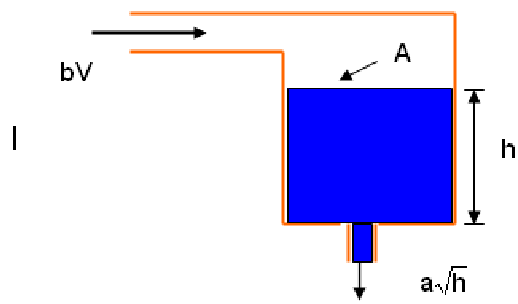


i(0) = 0;   h(0)=0.05;   h'(0)=0


altura vs tiempo
0.075
0.07
0.065
0.06
0.055
0.05
0.045
0
2
4
6
8
10


voltaje vs tiempo
25
20
15
10
5
0
0
2
4
6
8
10

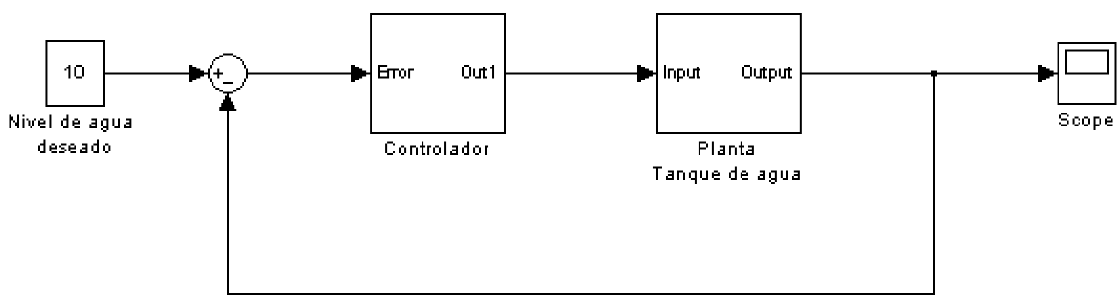
EJERCICIO4: TANQUE DE AGUA



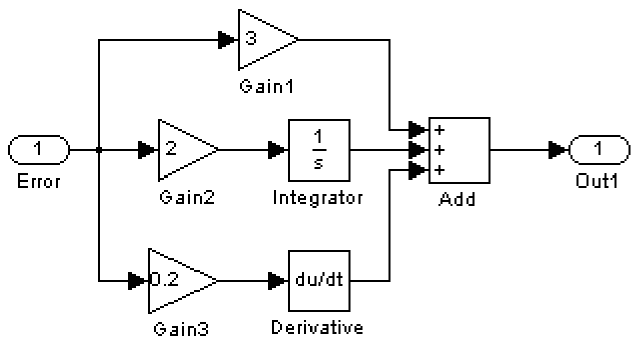
Ecuación del modelo:

$$\frac{dVol}{dt} = A\frac{dh}{dt} = bV - a\sqrt{h}$$

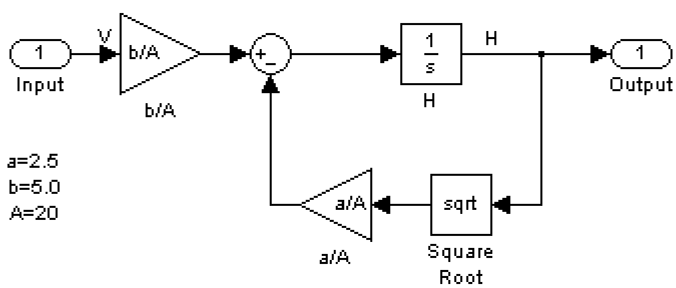
Diagrama simulink:



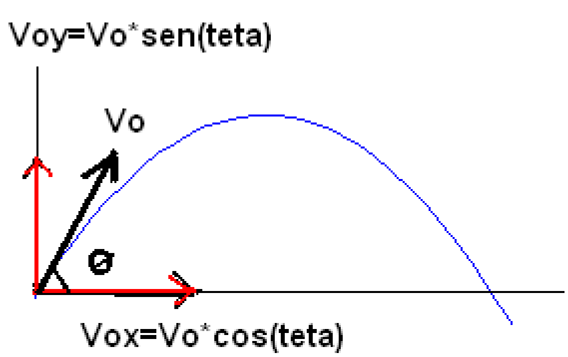
Controlador:



Planta:



EJERCICIO5: MOVIMIENTO PARABÓLICO



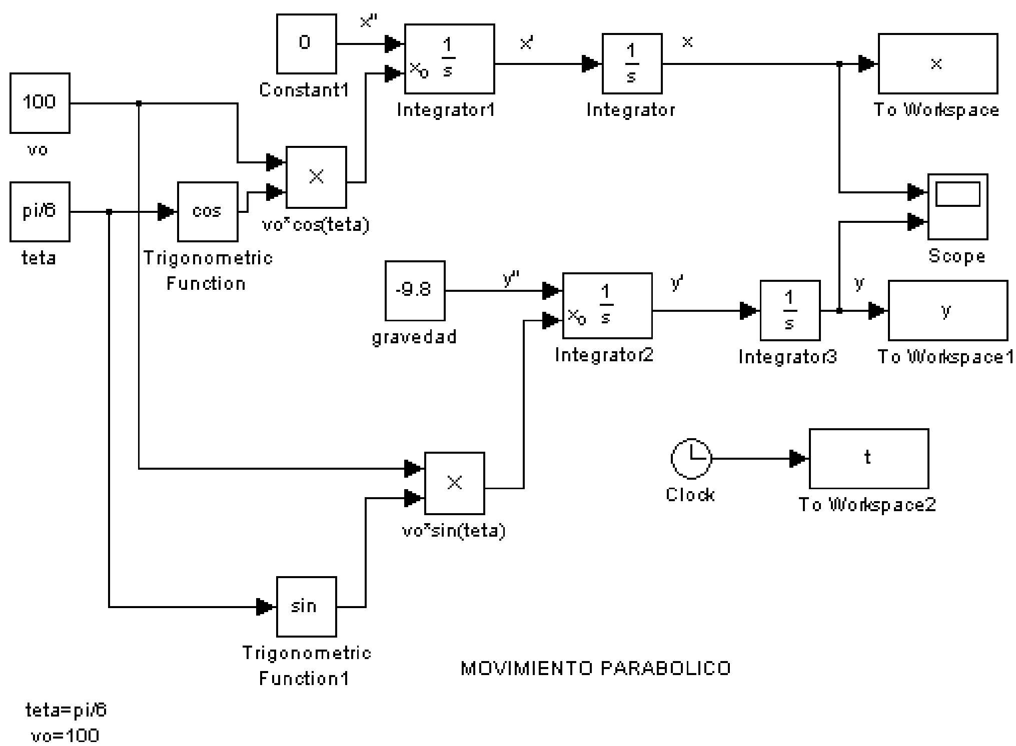
Ecuaciones:

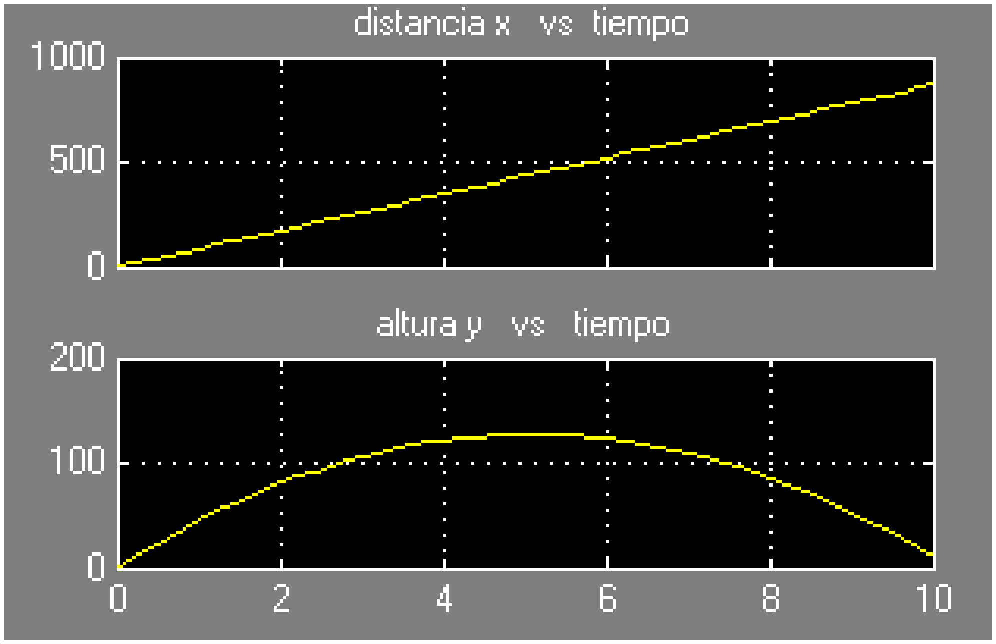
$x'' = 0$ Movimiento unforme

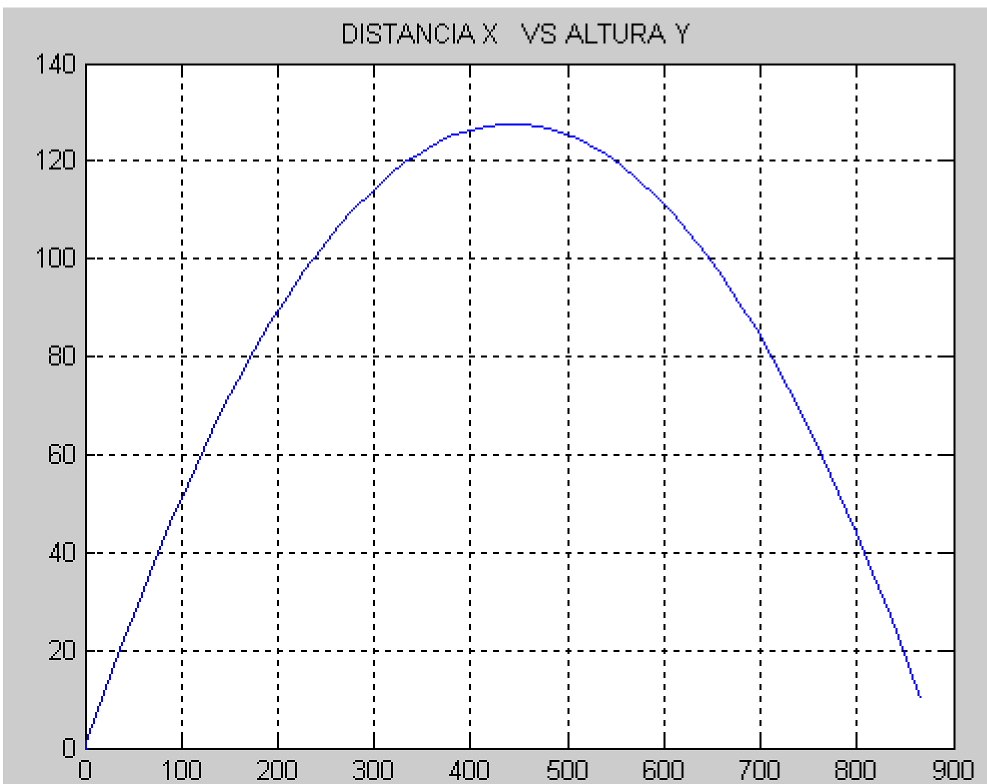
$y'' = -g$ Movimiento acelerado

Condiciones iniciales:

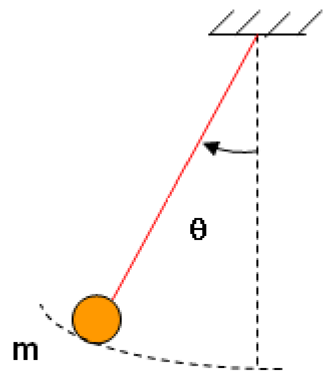
Vo=100 m/sg; $\theta = 30º$




distancia x vs tiempo
1000
500
0
altura y vs tiempo
200
100
0
0
2
4
6
8
10


DISTANCIA X VS ALTURA Y
140
120
100
80
60
40
20
0
0
100
200
300
400
500
600
700
800
900

EJERCICIO6: PÉNDULO SIMPLE



Ecuación:

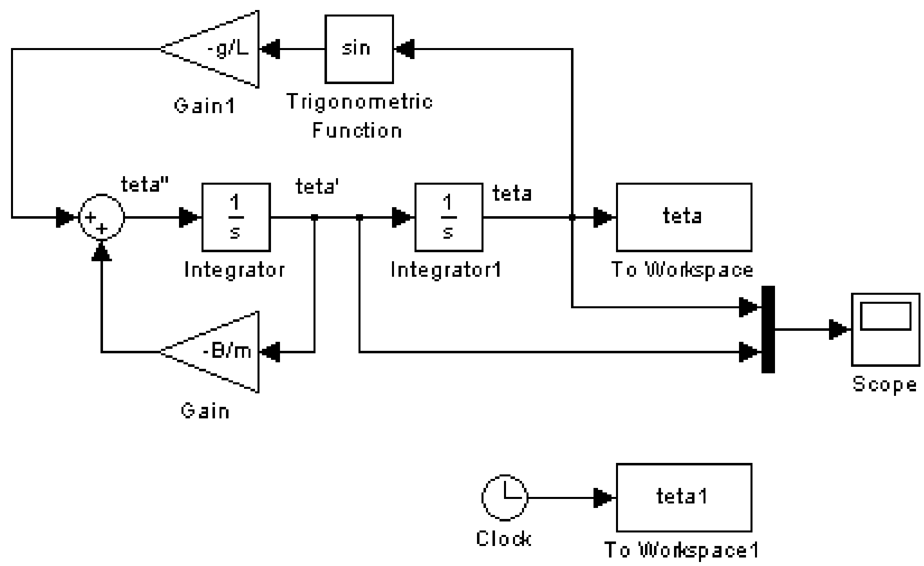
$$mL\theta'' + BL\theta' + wsen\theta = 0$$

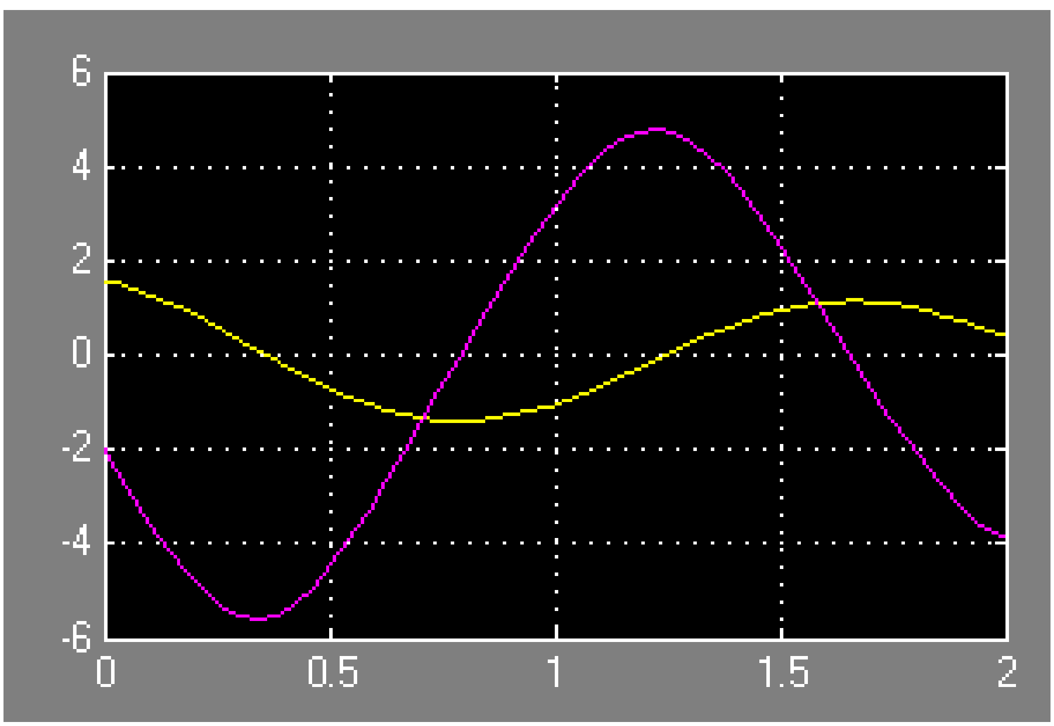
Valores:

w (peso) = 2; L (longitud) = 0.6; B (amortiguación) = 0.08;

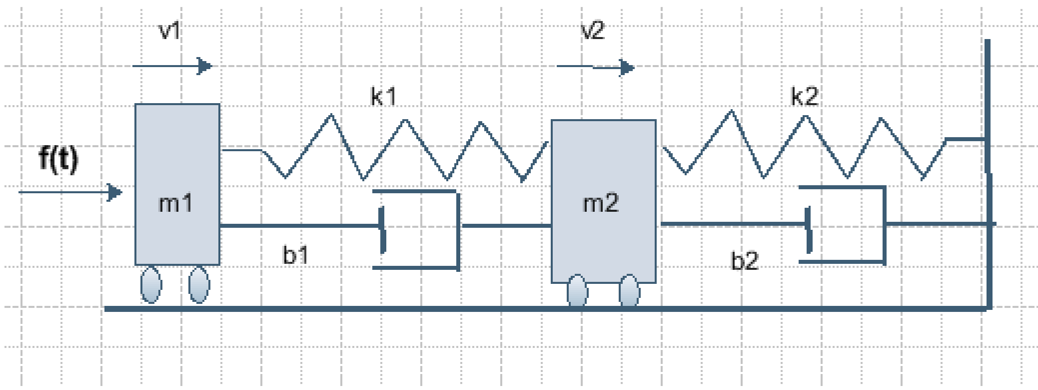
Condiciones iniciales: $\theta'(0) = -2$ rad/sg; $\theta(0) = \pi/2$

Diagrama simulink:





EJEMPLO: SISTEMA MECANICO



Parámetros:

m1=40; m2=60; k1=400; k2=400; b1=180; b2=220;

Ecuaciones dinámicas:

$$f(t) = m1\frac{dv1}{dt} + k1\int(v1 - v2)dt + (v1 - v2)b1$$

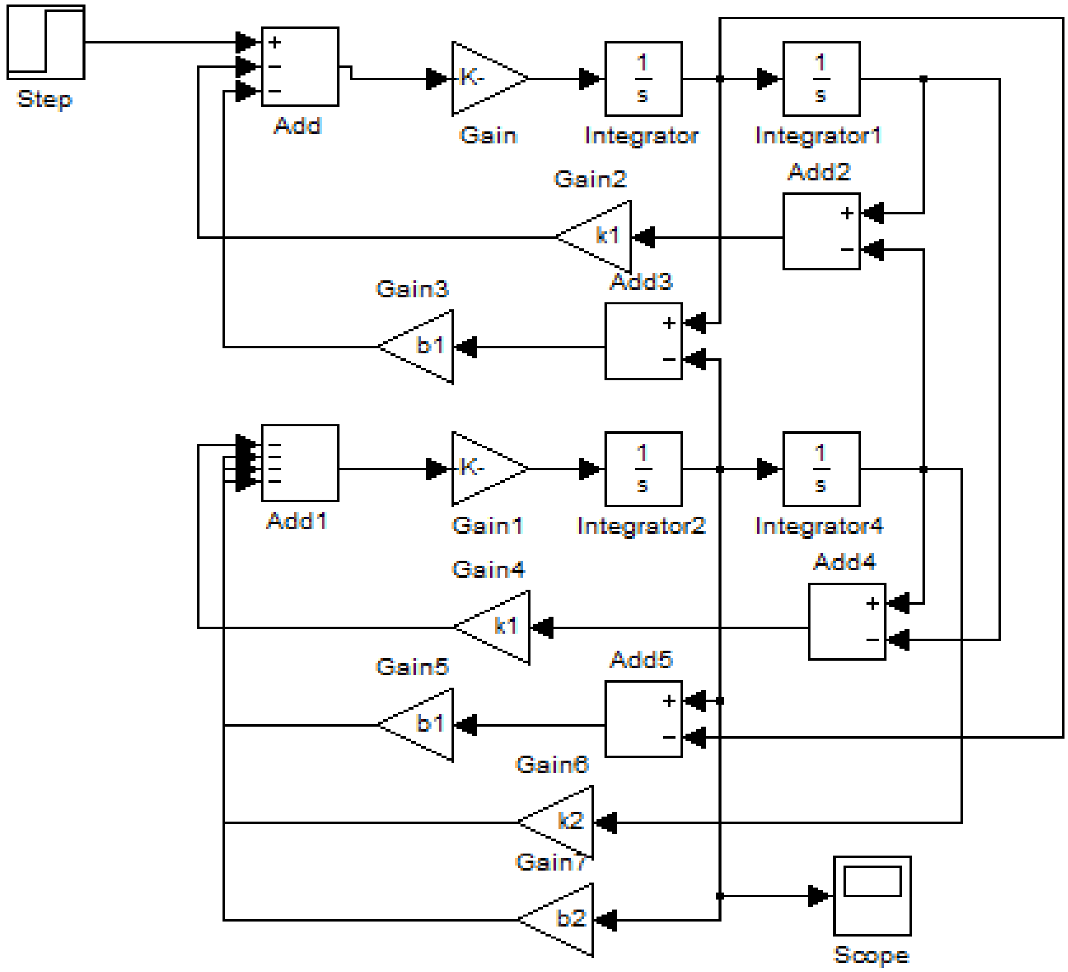
$$0 = m2\frac{dv2}{dt} + k1\int(v2 - v1)dt + (v2 - v1)b1 + k2\int v2dt + b2v2$$

Ecuaciones de Laplace:

$$F(s) = m1sV1 + \frac{K1}{s}(V1 - V2) + (V1 - V2)b1$$

$$0 = m2sV2 + \frac{k1}{s}(V2 - V1) + (V2 - V1)b1 + \frac{k2}{s}V2 + b2V2$$

Diagrama simulink:



EJEMPLO: SISTEMA TERMOQUÍMICO

Se desarrolla una reacción termoquímica en donde el reaccionante A se convierte en un producto B.

Velocidad de reacción: $r(t)= k\, c(t)$

Constante de velocidad de reacción: k = 0,2 min^{-1}

Concentración de la entrada: $c_i(t)$

Para t= 0; $c_i(0)$=1.25 lbmol/pie^3

Volumen de la masa reaccionante: V= 5 litros

Flujo de entrada: F= 1 lt/min

Ecuación dinámica:

$$V\frac{dc(t)}{dt} = Fc_i(t) - Fc(t) - KVc(t)$$

$$V\frac{dc(t)}{dt} = Fc_i(t) - (F + KV)c(t)$$

$$\frac{V}{F + KV}\frac{dc(t)}{dt} + c(t) = \frac{F}{F + KV}c_i(t)$$

Constante de tiempo:

$$\tau = \frac{V}{F + KV}$$

Ganancia de estado estacionario:

$$K_e = \frac{F}{F + KV}$$

Reemplazando valores: τ = 2.5 min; K_e = 0.5;

Condición inicial de la concentración: c(0)

0 = $Fc_i(0)$ - Fc(0) - KVc(0)

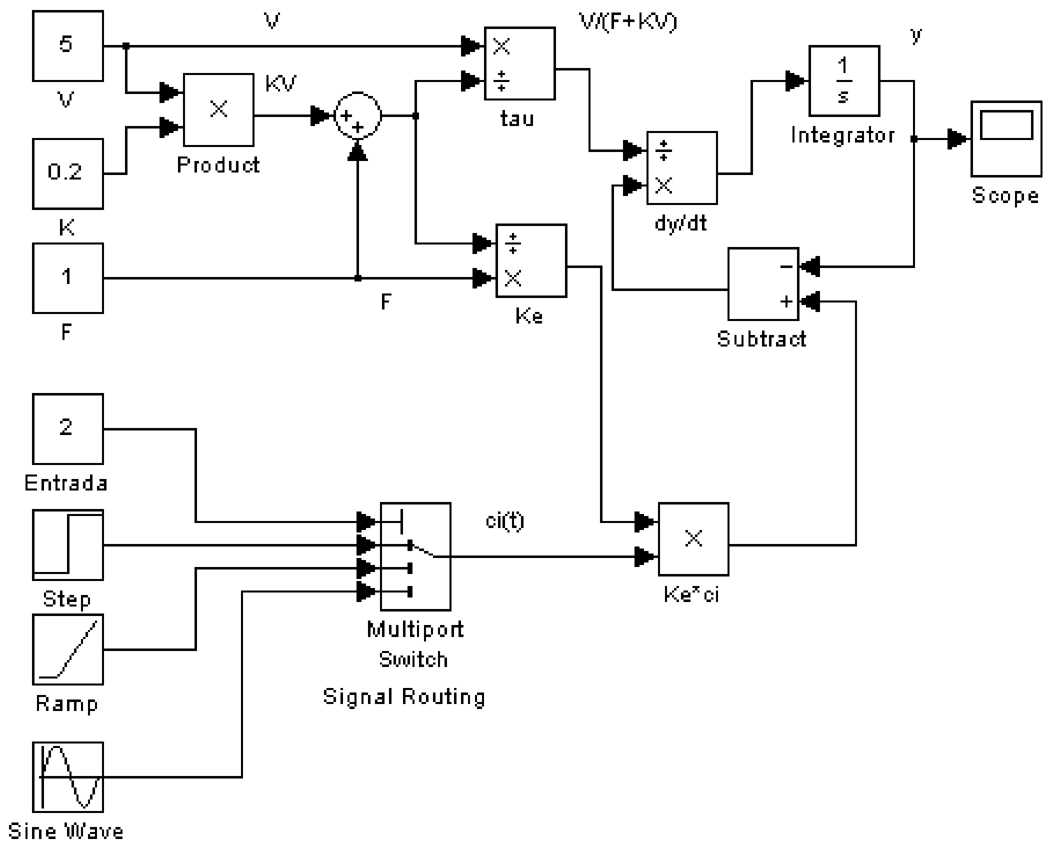
Reemplazando valores: c(0) = 0.625 lbmol/pie^3

Programa en Matlab:

```
%Entrada al paso. Programa pplineal.m
function dy=pplineal(t,y)
global K X tau
dy=(K*X-y)/tau;

% Entrada rampa. Programa rplineal.m
function dy=rplineal(t,y)
global K r tau
dy=(K*r*t-y)/tau;

% Entrada senoidal. Programa splineal.m
function dy=splineal(t,y)
global K tau A w
dy=(K*A*sin(w*t)-y)/tau;

% Programa principal
F=1;
V=5;
K=0.2;
ci0=1.25;
c0=solve('F*ci0-F*c0-K*V*c0=0');
c0=eval(c0)
%Constante de tiempo
tau=V/(F+K*V)
% tau=2.5 minutos
%Ganancia en estado estacionario
Ke=F/(F+K*V)
% Ke=0.5
```

```
global R K tau X r A w Rango Inicio
Rango=input('Tiempo de simulacion=');
Inicio=input('Condiciones iniciales=');
N=input('ESCRIBA 1=PASO, 2=RAMPA, 3=SENO:  ');
disp(' ')
switch N
   case 1
      X=input('Valor del paso=');
      [t,y]=ode45('pplineal',Rango,Inicio);
      plot(t,y)

 case 2
      r=input('valor pendiente de la rampa=');
      [t,y]=ode45('rplineal',Rango,Inicio);
      plot(t,r*t,t,y/K,'r')

   case 3
      A=input('Amplitud del seno=');
      w=input('Frecuencia del seno=');
      [t,y]=ode45('splineal',Rango,Inicio);
      disp('Amplitud del perfil de la respuesta')
      K*A/sqrt(1+(w*tau)^2)
      disp('Fase de la respuesta respecto a la entrada')
      atan(-w*tau)
      plot(t,A*sin(w*t),t,y,'r')
end
```

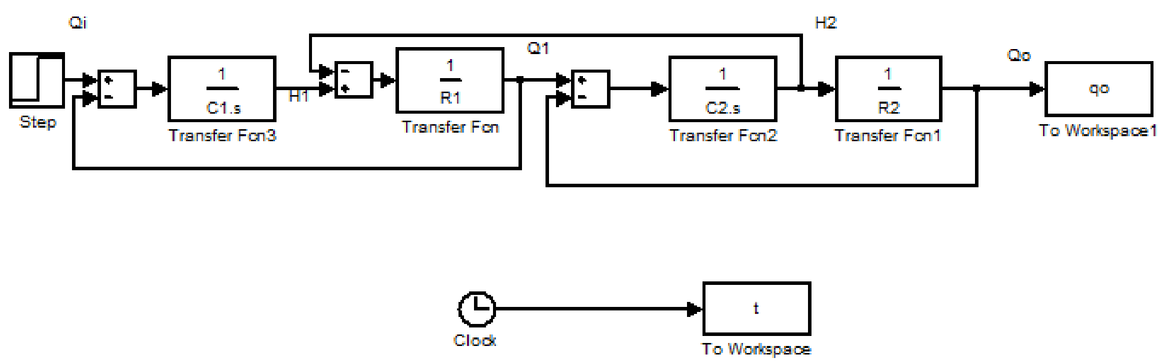
Programa en Simulink:



EJEMPLO: SISTEMA HIDRAULICO

```
% HIDRAULICO UNA ETAPA
C1=3; R1=1; C2=10; R2=2;
qi=2;
keyboard
plot(t,qo)
title('HIDRAULICO')
grid
```

```
pause
n=1;
while n==1
   T=input('Entre tiempo:  ')
   delta=input('Entre valor delta:  ')
   i=find(t<=(T+delta)&t>=(T-delta));
   tiempo=t(i)
   caudal_salida=qo(i)
   n=input('Entre 1 para seguir y 0 para parar:  ')
end
```



EJEMPLO: SISTEMA ELÉCTRICO

```
% CIRCUITO RC DE DOS ETAPAS
R1=10e3; R2=20e3; C1=1e-6; C2=10e-6;
ei=10;
keyboard
plot(t,eo)
title('CIRCUITO RC')
grid
```

```
pause
n=1;
while n==1
T=input('Entre tiempo:  ')
delta=input('Entre valor de delta:  ')
i=find(t<=(T+delta)&t>=(T-delta));
tiempo=t(i)
voltaje_salida=eo(i)
n=input('Entre 1 para seguir y 0 para parar:  ')
end
```