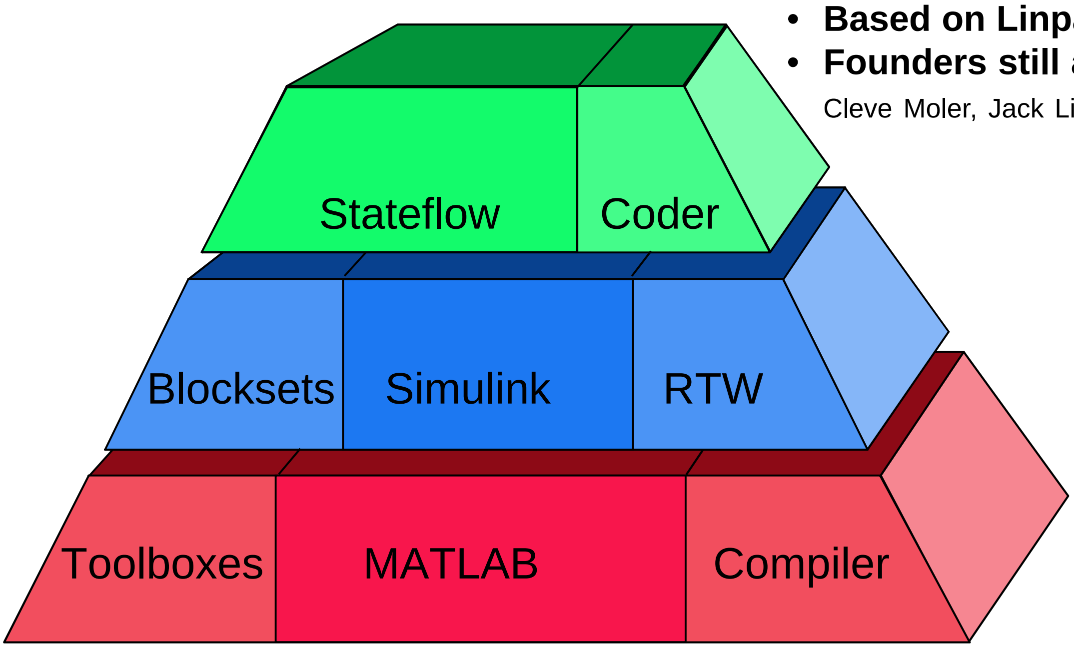
A Mini course on Dynamic Simulation (MATLAB and SIMULINK)

陳榮輝　中原大學化工系
e-mail : jason@wavenet.cycu.edu.tw
http://wavenet.cycu.edu.tw/

The MATLAB Product Family

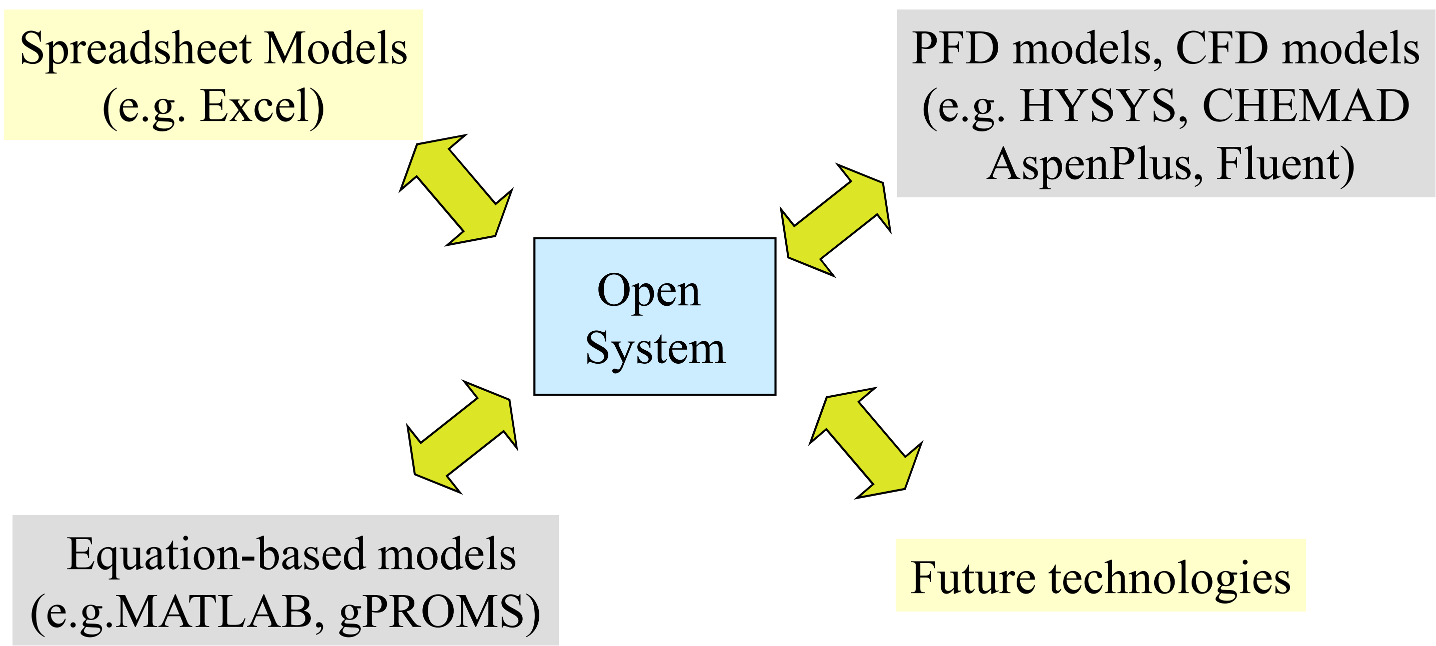
- **Founded in 1984**
- **Based on Linpack & Eispack**
- **Founders still active:**
  Cleve Moler, Jack Little, Steve Bangert



Pioneers in interactive technical software

Dynamic Simulation

Chemical Process Simulation



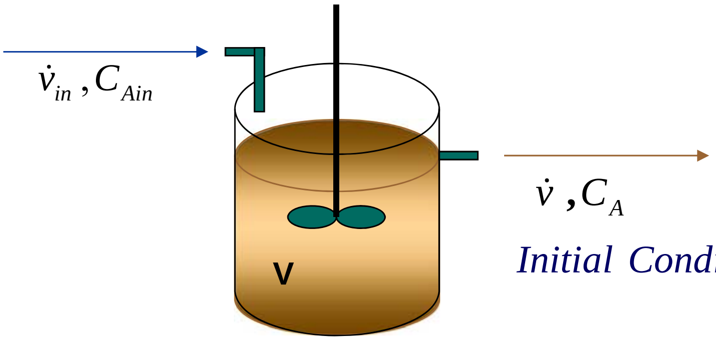
Example: Single-Variable Differential Equation

Objective: Solve differential mole balance on a CSTR using MATLAB integration routine.

Problem description: A CSTR initially filled in 2mol/L of A is started up with specified conditions of inlet concentration, inlet and outlet flow rates. The reactor volume and fluid density are considered to be constant.

$$\frac{dC_A}{dt} = -\left(\frac{\dot{v}}{V} + k\right)\cdot C_A + \frac{\dot{v}_{in}}{V}\cdot C_{Ain}$$

$$\dot{v} = \dot{v}_{in}$$



Reaction: A → B

Rate Kinetics: $(-r_A) = k\cdot C_A$

Initial Condition: at t=0, $C_A = C_{A,initial}$ = 2 mol/L

Ordinary Differential Equations: IVP

$$\frac{dy(t)}{dt} = f\left(t, y(t)\right)$$

$$y(t_0) = y_0$$

- MATLAB has several routines for numerical integration ode45, ode23, ode113, ode15s, ode23s, etc.
- Here we will introduce ode45 only.
- ode23 uses 2nd-order and ode45 uses 4th-order Runge-Kutta integration.

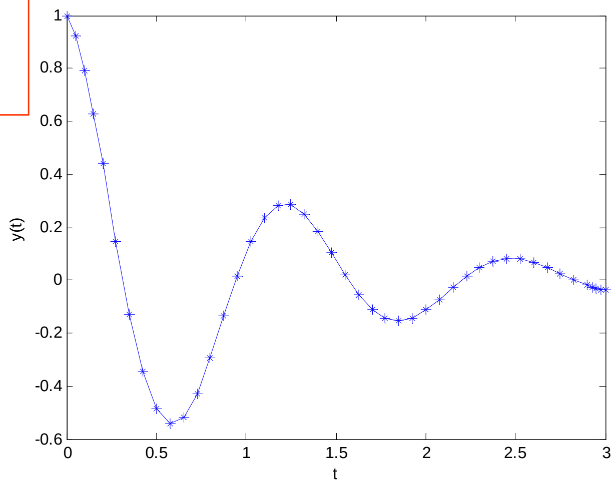
Example: Scalar ODE

$$\frac{dy(t)}{dt} = -y(t) - 5e^{-t}\sin 5t$$

$$y(0) = 1$$

```
function yprime = myf(t,y)
%MYF ODE example function
%    YPRIME = MYF(T,Y) evaluates derivatives

yprime = -y - 5*exp(-t)*sin(5*t);
```

```
% Case A
tspan = [0 3]; yzero = 1;
[t, y] = ode45(@myf,tspan,yzero);
plot(t,y,'*--');
xlabel t; ylabel y(t)
```



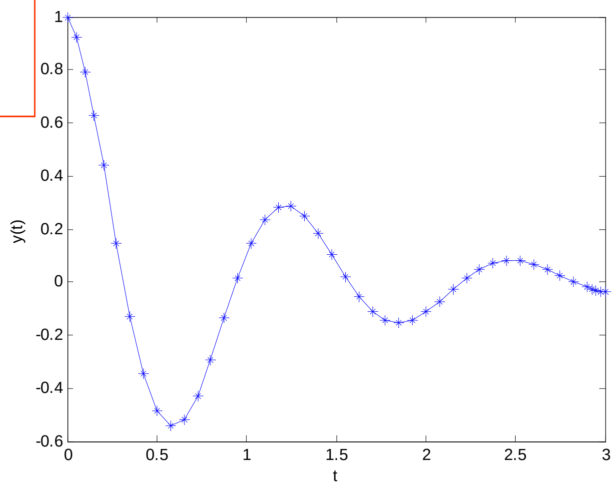
Example 1: Scalar ODE

$$\frac{dy(t)}{dt} = -y(t) - 5e^{-t}\sin 5t$$

$$y(0) = 1$$

```
function yprime = myf(t,y)
%MYF ODE example function
%    YPRIME = MYF(T,Y) evaluates derivatives

yprime = -y - 5*exp(-t)*sin(5*t);
```

```
>> tspan2 = 0:4;
>> [t2, y2] = ode45(@myf,tspan2,yzero);
>> disp([t2 y2])
         0    1.0000
    1.0000    0.1043
    2.0000   -0.1136
    3.0000   -0.0378
    4.0000    0.0075
```



Integration by ode23 and ode45

[t, y] = ode45 (‘@fcn’, [t0,tf], y0)

where

@fcn	is a string variable containing the name of the m-file for the derivatives.
t0	is the initial time
tf	is the final time
y0	is the initial condition vector for the state variables
t	a (column) vector of time
y	an array of state variables as a function of time

Example: Multivariable ODE

$$\frac{d^2\theta(t)}{dt^2} + \sin\theta(t) = 0$$

The first important point to note is that y is a vector of 2 variables, y1 and y2

Def:

$$y_1(t) = \theta(t)$$

$$y_2(t) = \frac{d\theta(t)}{dt}$$

$$\frac{dy_1(t)}{dt} = y_2(t)$$

$$\frac{dy_2(t)}{dt} = -\sin y_1(t)$$

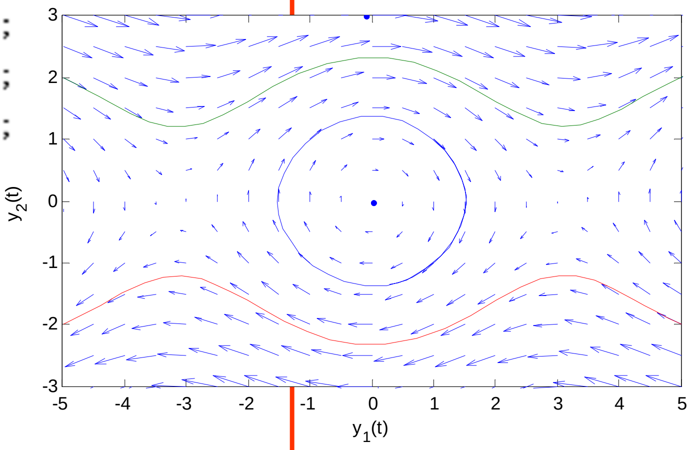
```
function dy = pend(t,y)
%PEND  Simple pendulum
%       DY = PEND(T,Y)
dy = [y(2); -sin(y(1))];
```

Also, dy is a vector of two differential equations associated with the 2 variables

Example: Multivariable ODE

$$\frac{d^2\theta(t)}{dt^2} + \sin\theta(t) = 0$$

```
tspan = [0 10];
yazero = [1; 1];ybzero = [-5; 2];yczero = [5; -2];

[ta, ya] = ode45(@pend,tspan,yazero);
[tb, yb] = ode45(@pend,tspan,ybzero);
[tc, yc] = ode45(@pend,tspan,yczero);

[y1 y2] = meshgrid(-5:.5:5,-3:.5:3);
Dy1Dt = y2; Dy2Dt = -sin(y1);
quiver(y1,y2,Dy1Dt,Dy2Dt)
hold on

plot(ya(:,1),ya(:,2),yb(:,1),yb(:,2),yc(:,1),yc(:,2))
axis equal; axis([-5 5 -3 3]);
xlabel y_1(t), ylabel y_2(t), hold off
```



Example: Rossler System

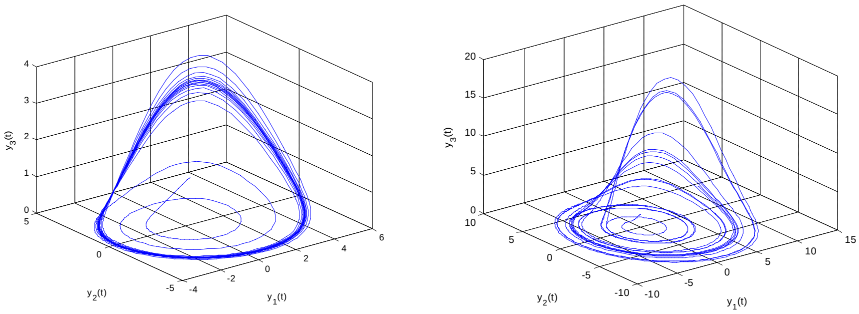
$$\frac{dy_1}{dt} = -y_2(t) - y_3(t)$$
$$\frac{dy_2}{dt} = y_1(t) + ay_2(t)$$
$$\frac{dy_3}{dt} = b + y_3(t)(y_1(t) - c)$$

a, b and c are problem parameters that are passed on the ode functions

```
function dy = rossler(t,y,a,b,c)
%ROSSLER    Rossler system, parameterized.
dy = [-y(2)-y(3); y(1)+a*y(2); b+y(3)*(y(1)-c)];
```

Example: Rossler System

$$\frac{dy_1}{dt} = -y_2(t) - y_3(t)$$
$$\frac{dy_2}{dt} = y_1(t) + a y_2(t)$$
$$\frac{dy_3}{dt} = b + y_3(t)(y_1(t) - c)$$



```
tspan = [0 100]; yzero = [1;1;1];
options = odeset('AbsTol',1e-7,'RelTol',1e-4);

a = 0.2; b = 0.2; c = 2.5;
[t,y] = ode45(@rossler,tspan,yzero,options,a,b,c);
subplot(221), plot3(y(:,1),y(:,2),y(:,3)),
xlabel y_1(t), ylabel y_2(t), zlabel y_3(t), grid

c = 5;
[t,y] = ode45(@rossler,tspan,yzero,options,a,b,c);
subplot(222), plot3(y(:,1),y(:,2),y(:,3)),
xlabel y_1(t), ylabel y_2(t), zlabel y_3(t), grid
```

option can be set via odeset

PID Feedback Control Simulation: MATLAB

```
ts=0.001;  %Sampling time
xk=zeros(2,1); e_1=0; u_1=0;
for k=1:1:2000
        time(k) = k*ts;
        yd(k)=0.50*sin(1*2*pi*k*ts);
        para=u_1;
        tSpan=[0 ts];
        [tt,xx]=ode45('PlantModel',tSpan,xk,[],para);
        xk = xx(length(xx),:);
        y(k)=xk(1);

        e(k)=yd(k)-y(k);de(k)=(e(k)-e_1)/ts;
        u(k)=20.0*e(k)+0.50*de(k);
        %Control limit
        if u(k)> 1.0,  u(k)= 1.0; end
        if u(k)< 0.0,  u(k)= 0.0; end

        u_1=u(k);e_1=e(k);
end
```

$$u(k) = u_s + k_p e(k) + k_i \sum_{i=0}^{k} e(i) t_s + \frac{k_d}{t_s}\left(e(k) - e(k-1)\right)$$

»**M_DigitPID.m & M_Plant.m**

PID Feedback Control Simulation: MATLAB

```
figure(1);
plot(time,yd,'r',time,y,'k:','linewidth',2);
xlabel('time(s)');ylabel('yd,y');
legend('Ideal position signal','Position tracking');
figure(2);
plot(time,yd-y,'r','linewidth',2);
xlabel('time(s)'),ylabel('error');
```

Plant

$$G(s) = \frac{1}{Js^2 + Bs}$$

$$\frac{Y(s)}{U(s)} = \frac{1}{Js^2 + Bs} \Rightarrow J\frac{d^2 y(t)}{d^2 t} + B\frac{dy(t)}{dt} = u(t)$$

```
function dy = PlantModel(t,y,flag,para)
u=para;
J=0.0067;B=0.1;

dy=zeros(2,1);
dy(1) = y(2);
dy(2) = -(B/J)*y(2) + (1/J)*u;
```

$$y_1 = y$$

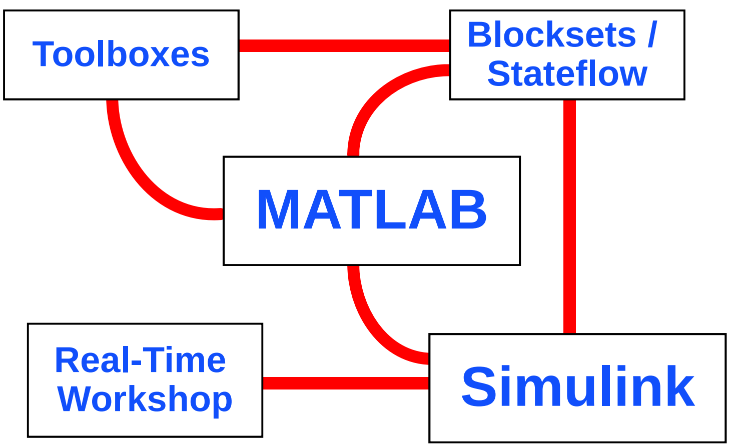
$$\frac{dy_1}{dt} = y_2$$

$$\frac{dy_2}{dt} = -\frac{B}{J}y_2 + \frac{1}{J}u$$

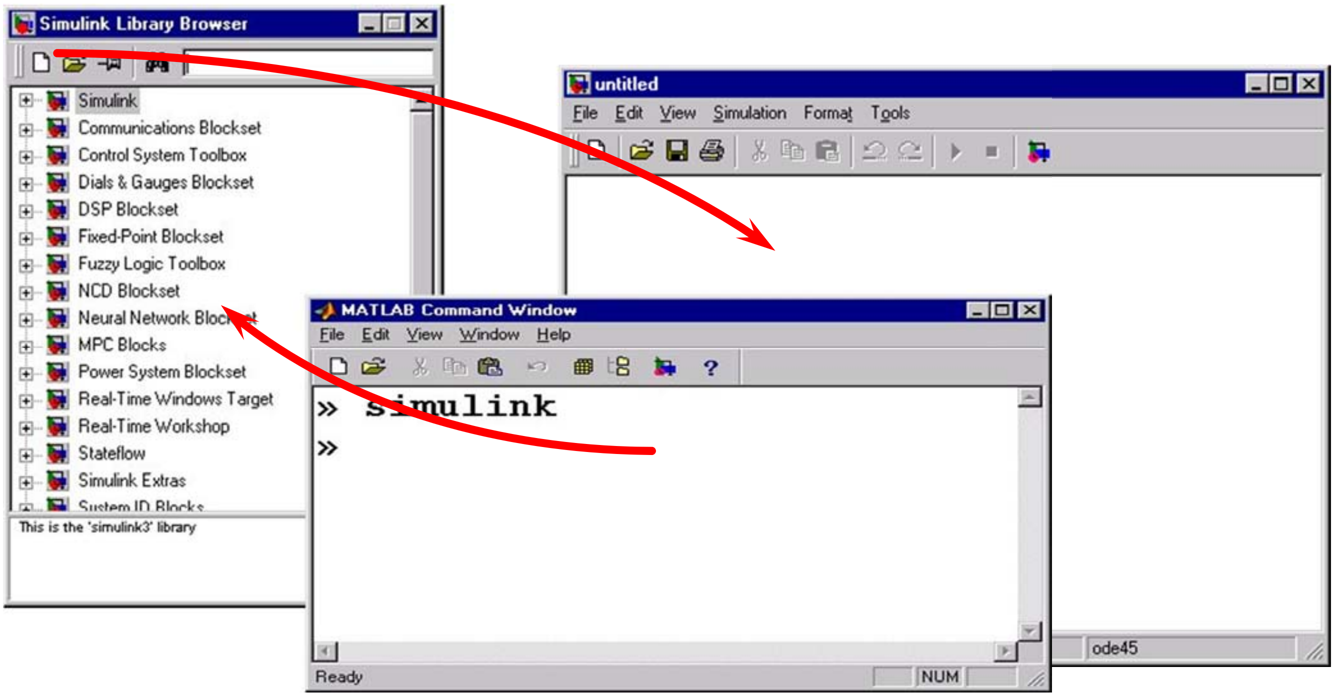
»M_DigitPID.m & M_Plant.m

What is SIMULINK

Simulink is an interactive, block-orientated-based tool. It is a block program that allows the simulation and analysis of dynamic systems in a block diagram format whether they are linear or nonlinear, in continuous or discrete forms.It is tightly coupled with MATLAB and supported by Blocksets and Extensions.



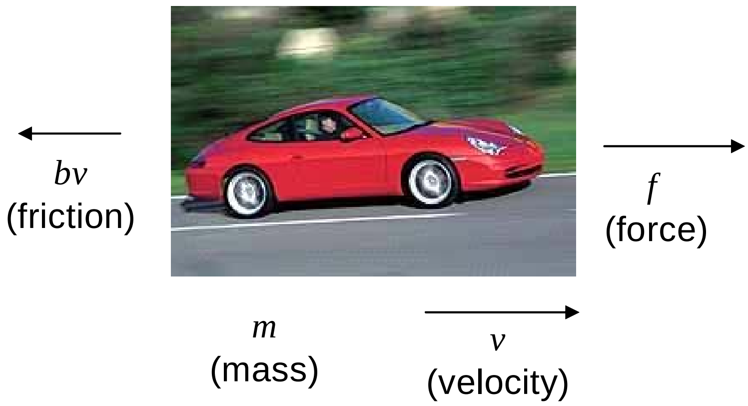
Starting SIMULINK



» simulink

EXAMPLE: SIMULINK the Motion of Car

In this example we will demonstrate what Simulink is capable of by simulating the motion of a car. We are interested in analyzing the relationship between the force applied to the car, f, and the resultant velocity, v, of the car.



$$m\frac{dv}{dt} = f - bv$$

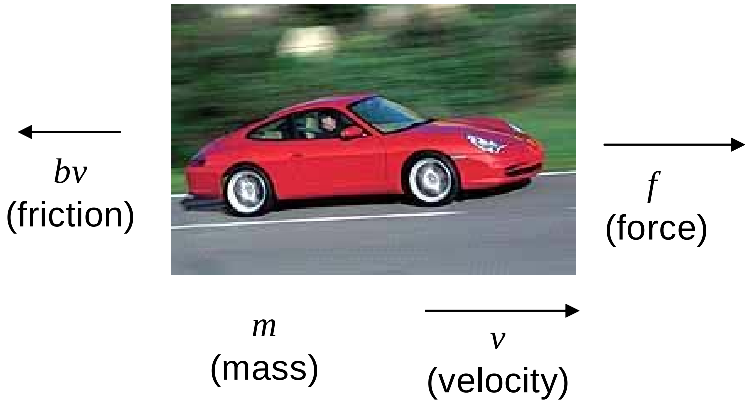
$$m = 1000kg \qquad b = 50Ns/m$$

Although it is possible to model the above system in Simulink directly, it is simpler if we convert the system into the Laplace domain.

$$m\frac{dv}{dt}+bv=f$$

$$msV(s)+bV(s)=F(s)$$

$$1000sV(s)+50V(s)=F(s)$$

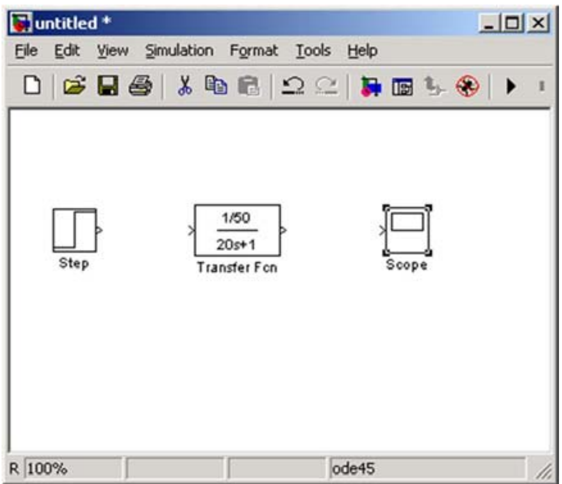


$$\frac{V(s)}{F(s)}=\frac{1}{1000s+50}$$

$$=\frac{\frac{1}{50}}{20s+1}\left(=\frac{K}{Ts+1}\right)$$

We will now find out what happens when we increase the force applied to the car from 0 N to 500 N.

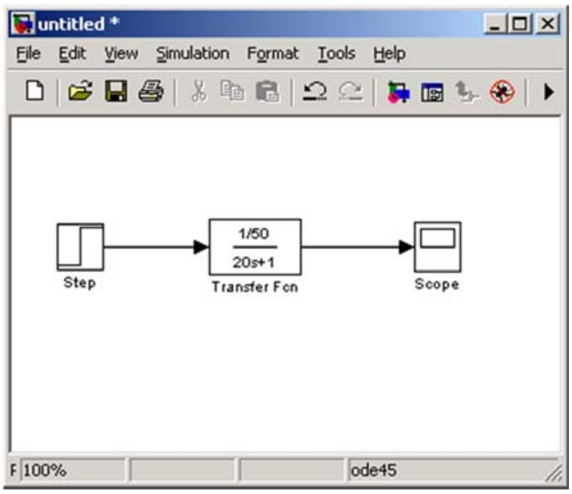
- **Sources → Step (a step time of 10, an initial value of 0 and a final value of 500.)**
- **Continuous → Transfer Fcn (the numerator should be equal to** [1/50] **and the denominator should be** [20 1]**.)**
- **Sinks → Scope**



EXAMPLE: SIMULINK the Motion of Car

We will now find out what happens when we increase the force applied to the car from 0 N to 500 N.

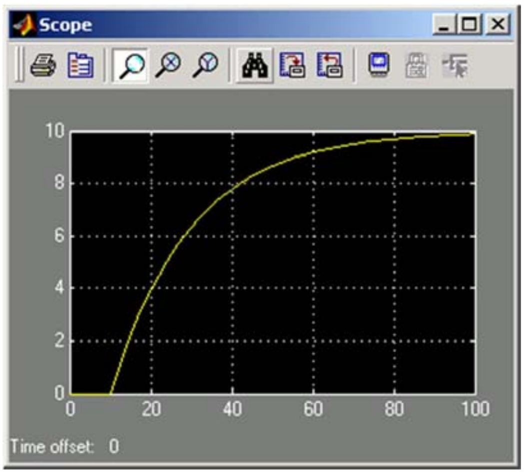
- **Sources → Step (a step time of 10, an initial value of 0 and a final value of 500.)**
- **Continuous → Transfer Fcn (the numerator should be equal to** [1/50] **and the denominator should be** [20 1]**.)**
- **Sinks → Scope**
- **Draw a line from the output of any block to the input of any other block.**
- **Select Simulation-Simulation Parameters. (In the stop time, enter a value of 100.)**
- **Double click on the scope block**
- **click on the play button (▸)**



EXAMPLE: SIMULINK the Motion of Car

We will now find out what happens when we increase the force applied to the car from 0 N to 500 N.

- **Sources → Step (a step time of 10, an initial value of 0 and a final value of 500.)**
- **Continuous → Transfer Fcn (the numerator should be equal to** [1/50] **and the denominator should be** [20 1]**.)**
- **Sinks → Scope**
- **Draw a line from the output of any block to the input of any other block.**
- **Select Simulation-Simulation Parameters. (In the stop time, enter a value of 100.)**
- **Double click on the scope block**
- **click on the play button**
- **Click on autoscale button to resize the scale to fit the entire range of vales**



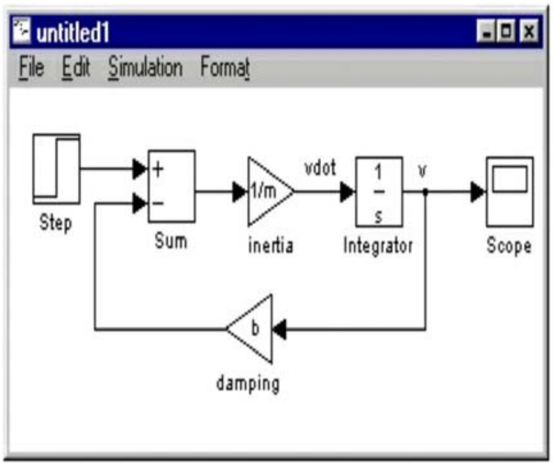
EXAMPLE: SIMULINK the Motion of Car

The simulink model could have been developed directly from the differential equation in the time domain.

$$m\frac{dv}{dt}+bv=f$$

$$m=1000 \quad b=50$$

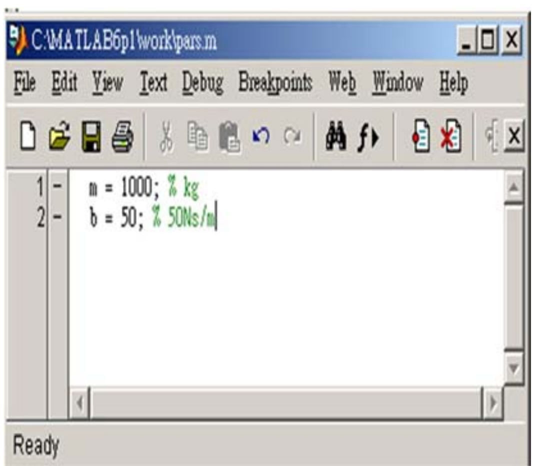
- The **inertia** and **damping** are ***gain*** blocks (located in the math folder). Double-click them, setting the parameter variables (m and b) (I prefer this way for easily checking your model).
- The damping block has been flipped around. To flip a block, right click on it and then select format-flip block from the pull down menu.
- The text for each of the blocks can be edited by simply clicking on the text and entering whatever you like
- The m-file is built up to edit the values of parameters.
- The **integration** is **integration** block (locate in the continuous folder). Double-click it, setting initial condition to 0 (default value).



EXAMPLE: SIMULINK the Motion of Car

The simulink model could have been developed directly from the differential equation in the time domain.

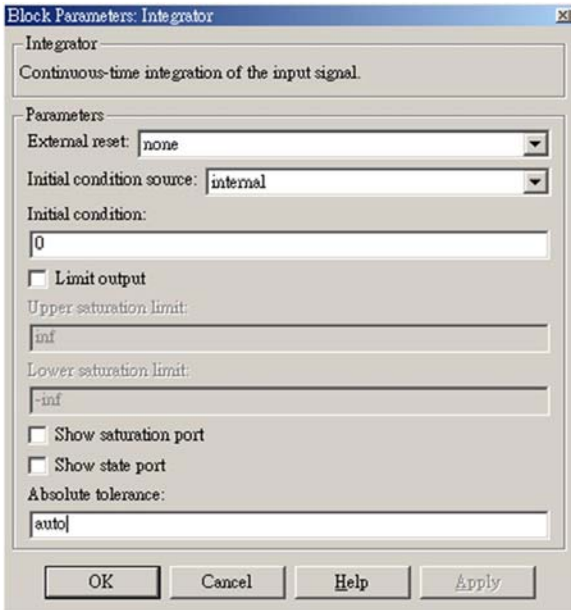
- The **inertia** and **damping** are ***gain*** blocks (located in the math folder). Double-click them, setting the parameter variables (m and b) (I prefer this way for easily checking your model).
- The damping block has been flipped around. To flip a block, right click on it and then select format-flip block from the pull down menu.
- The text for each of the blocks can be edited by simply clicking on the text and entering whatever you like
- The m-file is built up to edit the values of parameters.
- The **integration** is **integration** block (locate in the continuous folder). Double-click it, setting initial condition to 0 (default value).



EXAMPLE: SIMULINK the Motion of Car

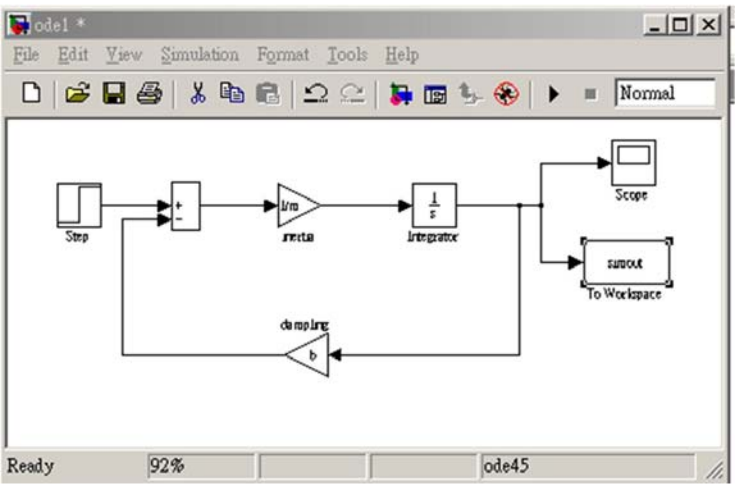
The simulink model could have been developed directly from the differential equation in the time domain.

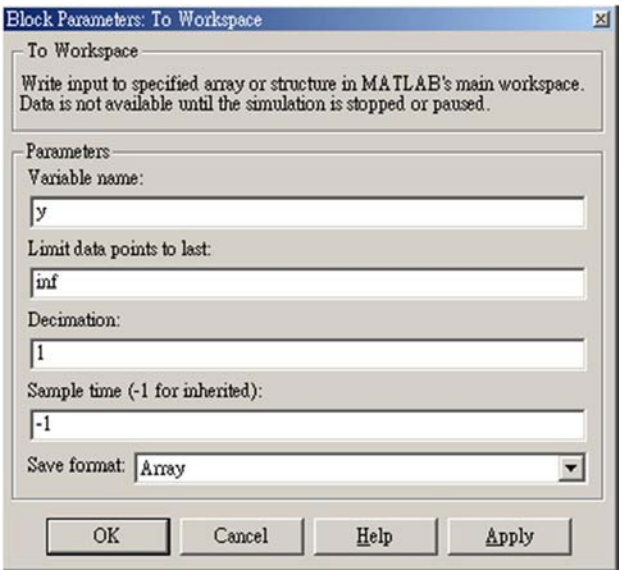
- **The inertia and damping are *gain* blocks (located in the math folder). Double-click them, setting the parameter variables (m and b) (I prefer this way for easily checking your model).**
- **The damping block has been flipped around. To flip a block, right click on it and then select format-flip block from the pull down menu.**
- **The text for each of the blocks can be edited by simply clicking on the text and entering whatever you like**
- **The m-file is built up to edit the values of parameters.**
- **The integration is integration block (locate in the continuous folder). Double-click it, setting initial condition to 0 (default value).**



EXAMPLE: SIMULINK the Motion of Car

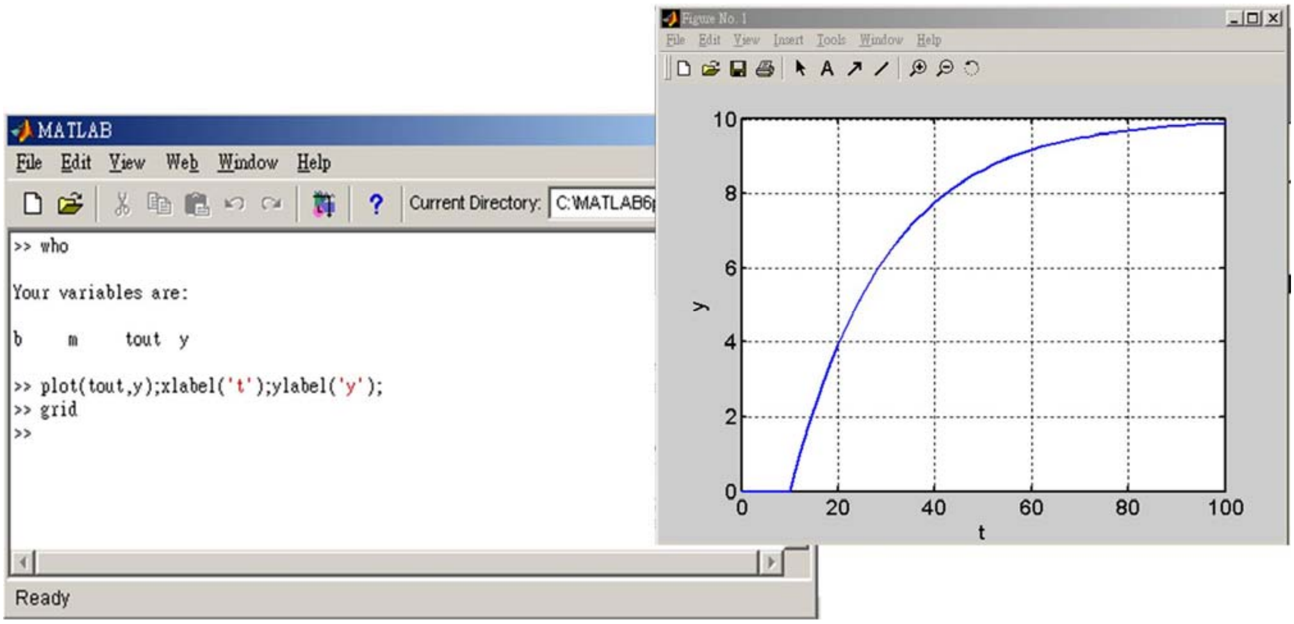
You can pass the data into the MATLAB workspace using the To Workspace icon





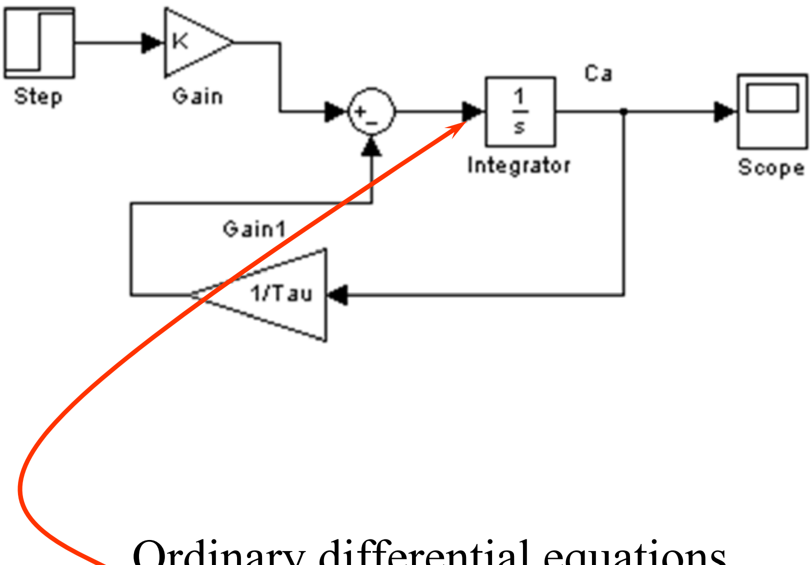
EXAMPLE: SIMULINK the Motion of Car

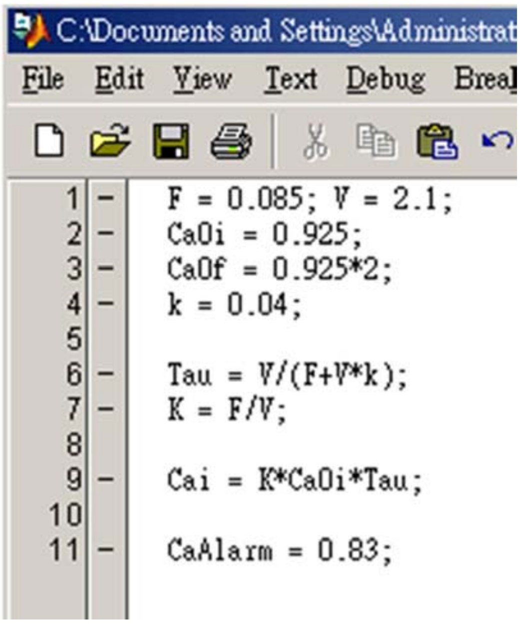
You can pass the data into the MATLAB workspace using the To Workspace icon



Exercise: Single Isothermal CSTR Reactor

$$\frac{dC_A}{dt} + \frac{1}{\tau} C_A = \frac{1}{\tau} C_{A0}$$





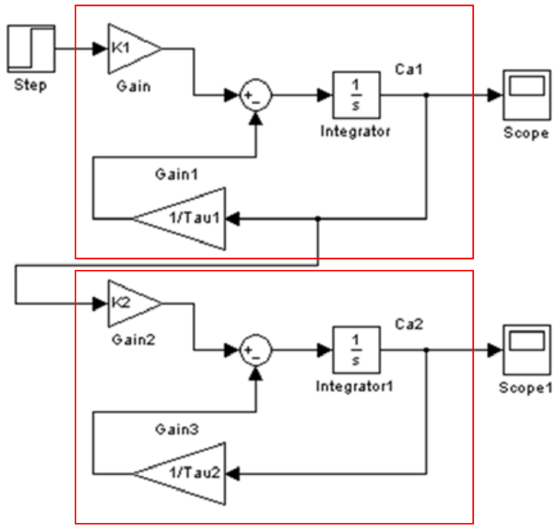

```
F = 0.085; V = 2.1;
Ca0i = 0.925;
Ca0f = 0.925*2;
k = 0.04;

Tau = V/(F+V*k);
K = F/V;

Cai = K*Ca0i*Tau;

CaAlarm = 0.83;
```

Ordinary differential equations

» **cstr1.mdl**

Exercise: Two Isothermal CSTR Reactors

$$1st \quad \text{tank}: \; V_1 \frac{dC_{A1}}{dt} = F\left(C_{A0} - C_{A1}\right) - V_1 k C_{A1}$$



$$2nd \quad \text{tank}: \; V_2 \frac{dC_{A2}}{dt} = F\left(C_{A1} - C_{A2}\right) - V_2 k C_{A2}$$

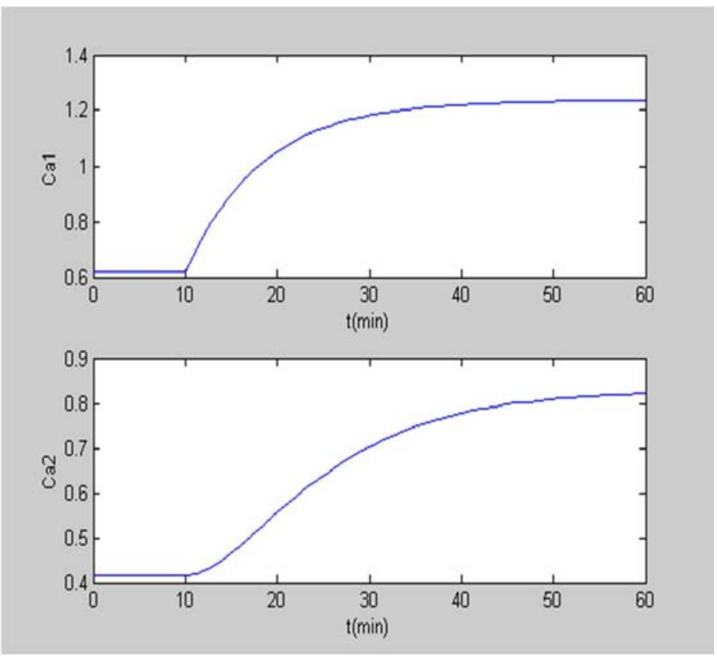
C:\Documents and Settings\Administrat

File Edit View Text Debug Breal

```
F = 0.085; V1 = 1.05;
Ca0i = 0.925;
Ca0f = 0.925*2;
k = 0.04;

Tau1 = V1/(F+V1*k);
K1 = F/V1;
Ca1i = Tau1*K1*Ca0i;

V2 = 1.05;
Tau2 = V2/(F+V2*k);
K2 = F/V2;
Ca2i = Tau1*K1*Ca1i;

CaAlarm = 0.83;
```

» **cstr2.mdl**

Exercise: Two Isothermal CSTR Reactors



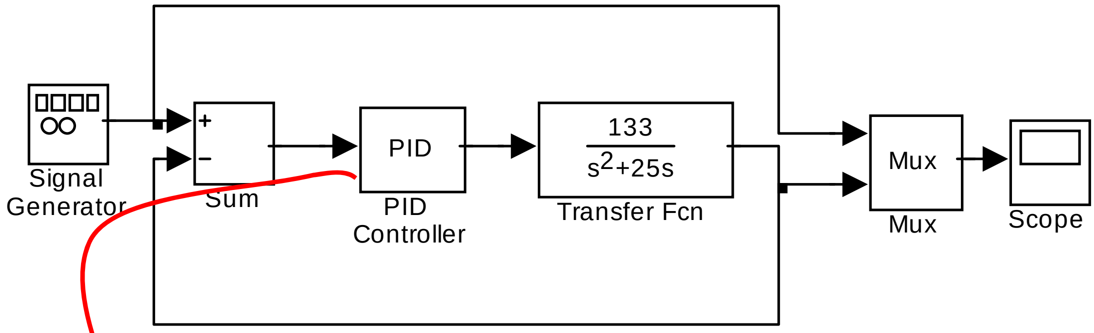
ory: D:\CYCUArchie\程序控制\Slides

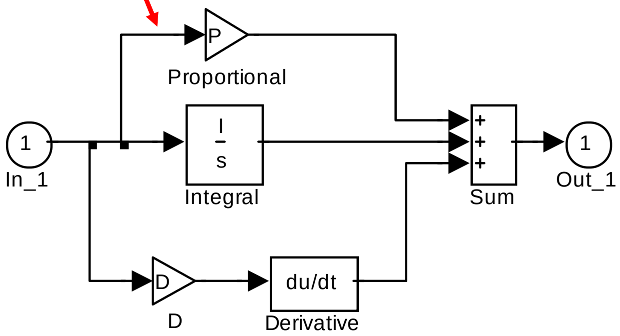
Command Window

```
>> who

Your variables are:

Ca0_init  Ca2_init  V1        deltCa0
Ca1_init  F         V2        k

>> who

Your variables are:

Ca0_init  Ca1_init  Ca2_init  V1        deltCa0   tout
Ca1       Ca2       F         V2        k

>> subplot(211)
>> plot(tout,Ca1);xlabel('t(min)');ylabel('Ca1')
>> subplot(212)
>> plot(tout,Ca2);xlabel('t(min)');ylabel('Ca2')
>>
```

» **cstr2.mdl**

PID Feedback Control Simulation: SIMULINK



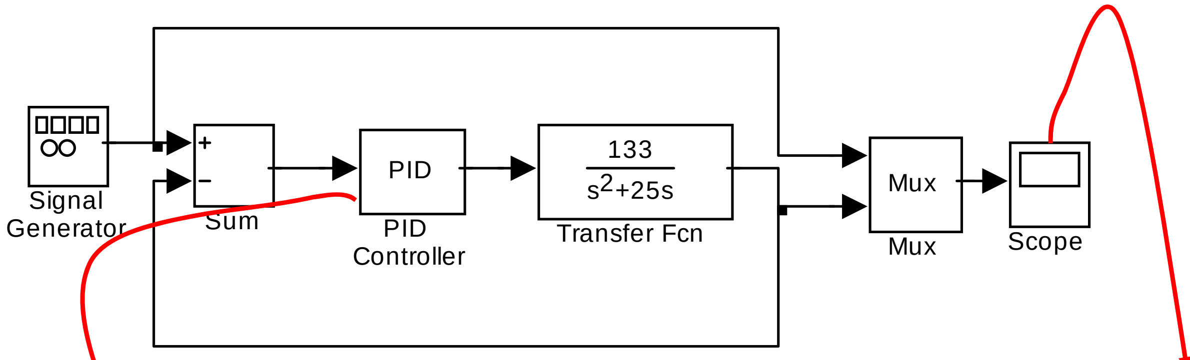


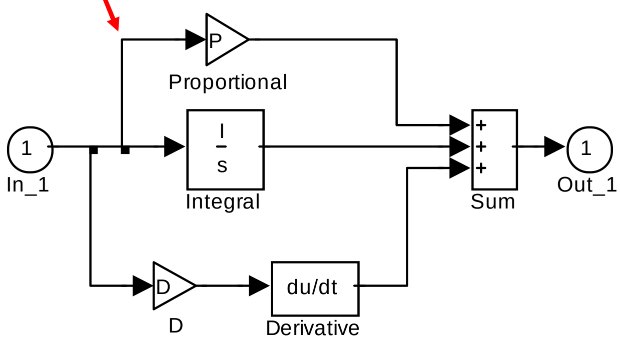
$$E(t) = SP(t) - CV(t)$$

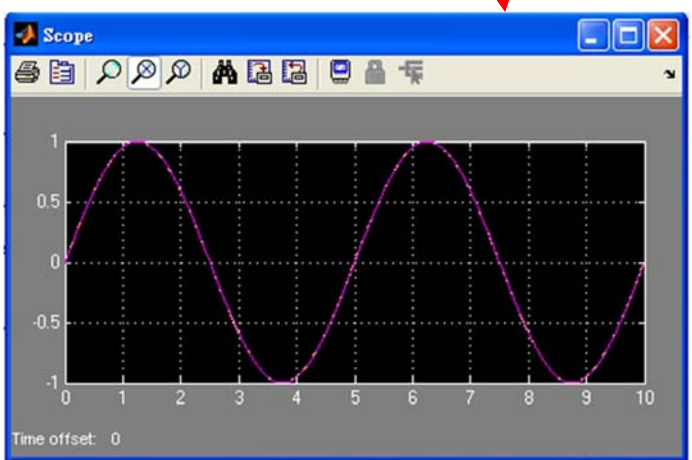
$$MV(t) = K_c \left[E(t) + \frac{1}{T_I} \int_0^t E(t')dt' + T_d \frac{dE}{dt} \right] + I$$

»SimLTI_PID1.mdl

PID Feedback Control Simulation: SIMULINK







»SimLTI_PID1.mdl