ME 1020 Engineering Programming with MATLAB

Chapter 10 Homework Solutions: 10.5, 10.8, 10.10, 10.12, 10.16, 10.19, 10.23, 10.30

Topics Covered:

- Simulation Diagrams
- Simulink Models
- Library Browser
- Commonly-Used Blocks
- Transfer-Function Models
- Linear State-Variable Models
- Piecewise-Linear Models
- Subsystems

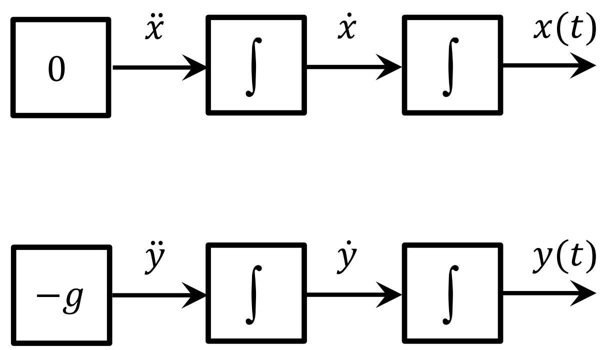
Problem 10.5:

5. A projectile is launched with a velocity of 100 m/s at an angle of 30° above the horizontal. Create a Simulink model to solve the projectile's equations of motion where x and y are the horizontal and vertical displacements of the projectile.

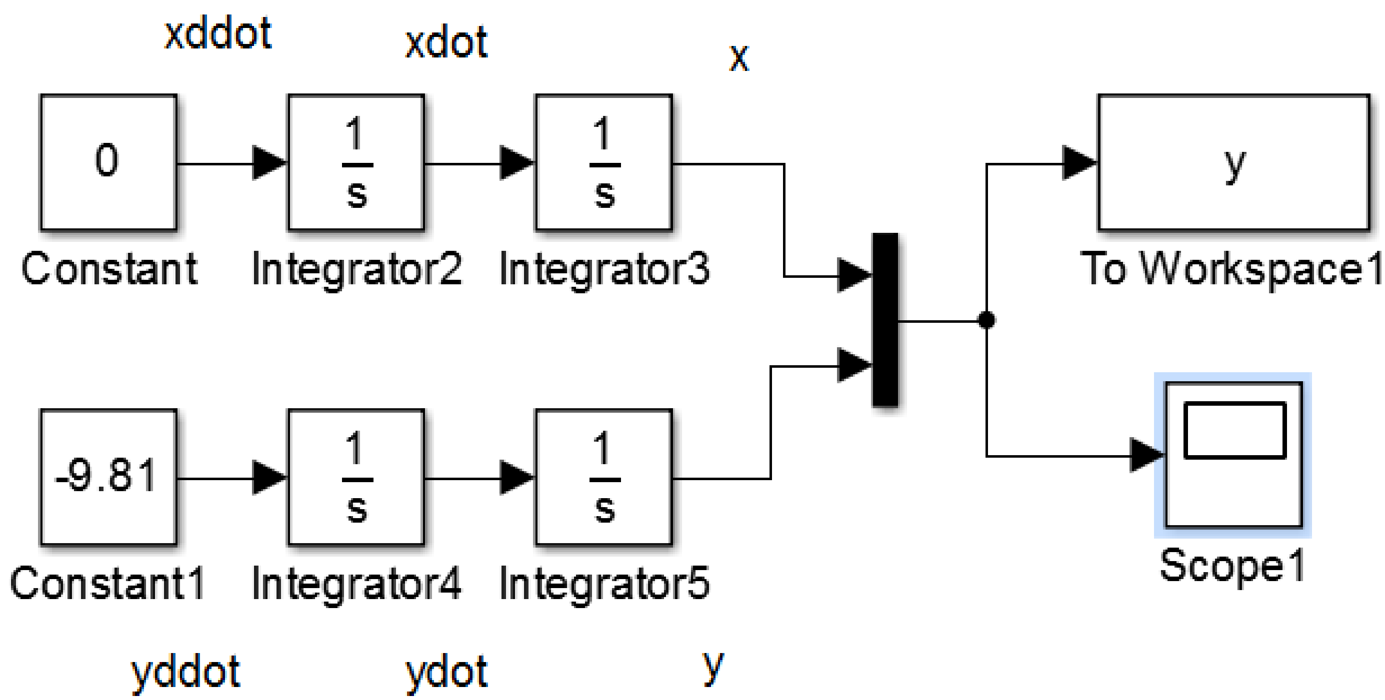
$$\ddot{x} = 0 \qquad x(0) = 0 \qquad \dot{x}(0) = 100 \cos 30^\circ$$
$$\ddot{y} = -g \qquad y(0) = 0 \qquad \dot{y}(0) = 100 \sin 30^\circ$$

Use the model to plot the projectile's trajectory y versus x for $0 \le t \le 10$ s.

Simulation Diagram:



Simulink Model:

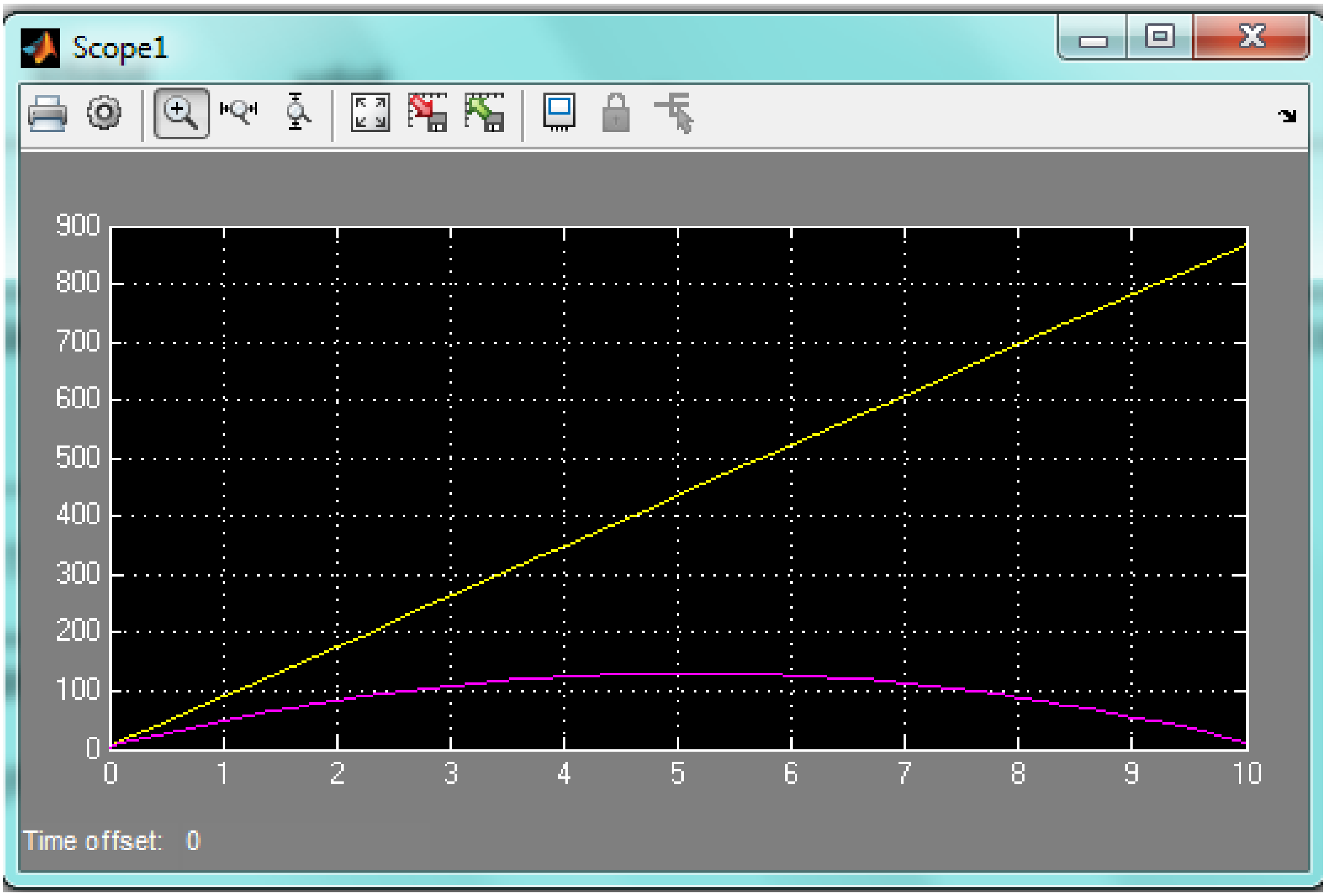


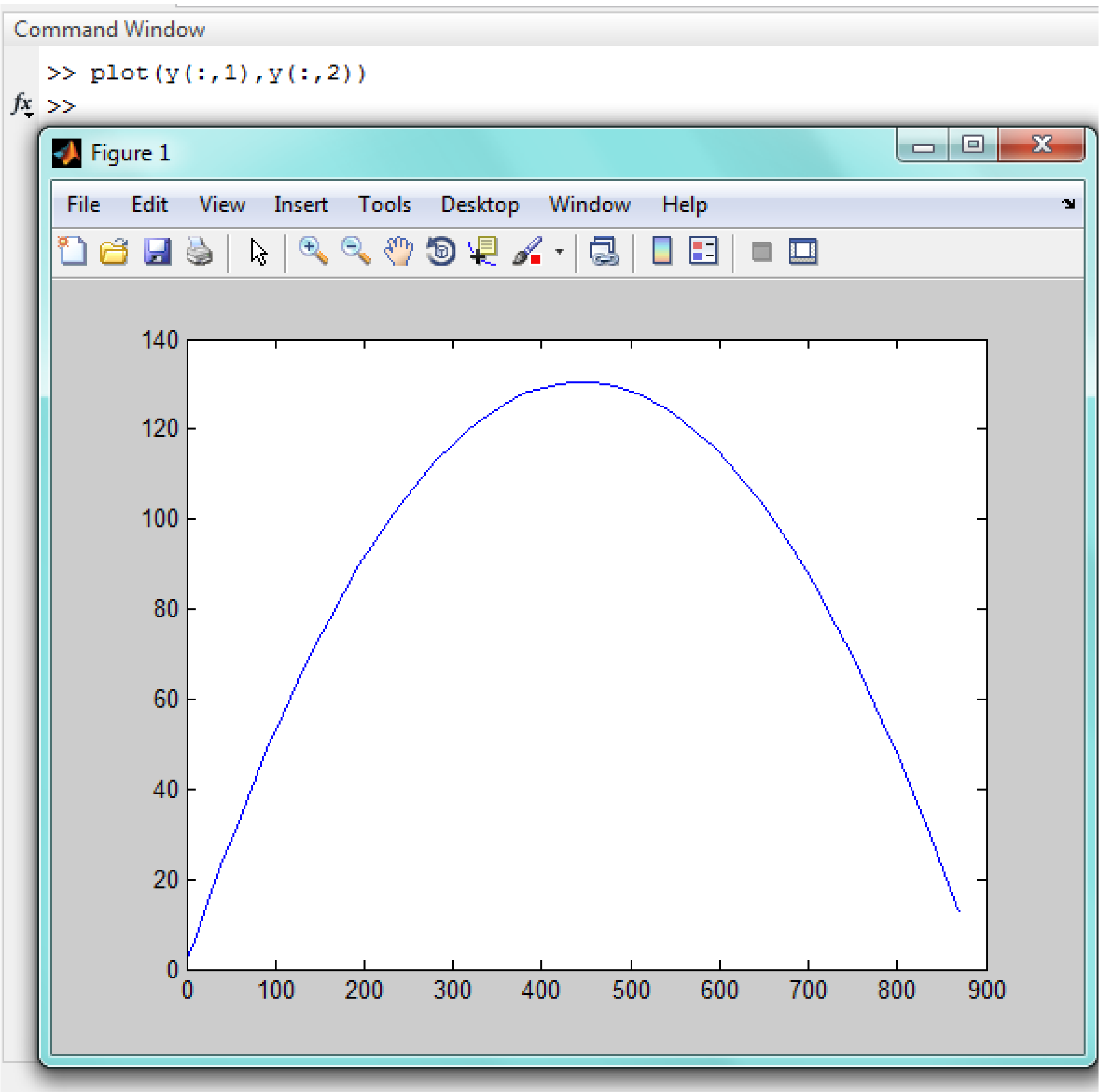
Note 1: Set the Initial Condition of the first x Integrator to 100 cos (30*pi/180): $[\dot{x}(0) = 100\cos(30°)]$ by double-clicking on the Integrator block. Set the Initial Condition of the first y Integrator to 100 sin (30*pi/180): $[\dot{y}(0) = 100\sin(30°)]$.

Note 2: Double-click on the To Workspace block. Set the Save format to Array.

Note 3: The Scope plot shows x and y versus t. Plot y versus x in the Command Window by typing:

```
>> plot(y(:,1),y(:,2))
```


Scope1
900
800
700
600
500
400
300
200
100
0
0
1
2
3
4
5
6
7
8
9
10
Time offset: 0



Command Window
>> plot(y(:,1),y(:,2))
>>
Figure 1
File Edit View Insert Tools Desktop Window Help
140
120
100
80
60
40
20
0
0 100 200 300 400 500 600 700 800 900

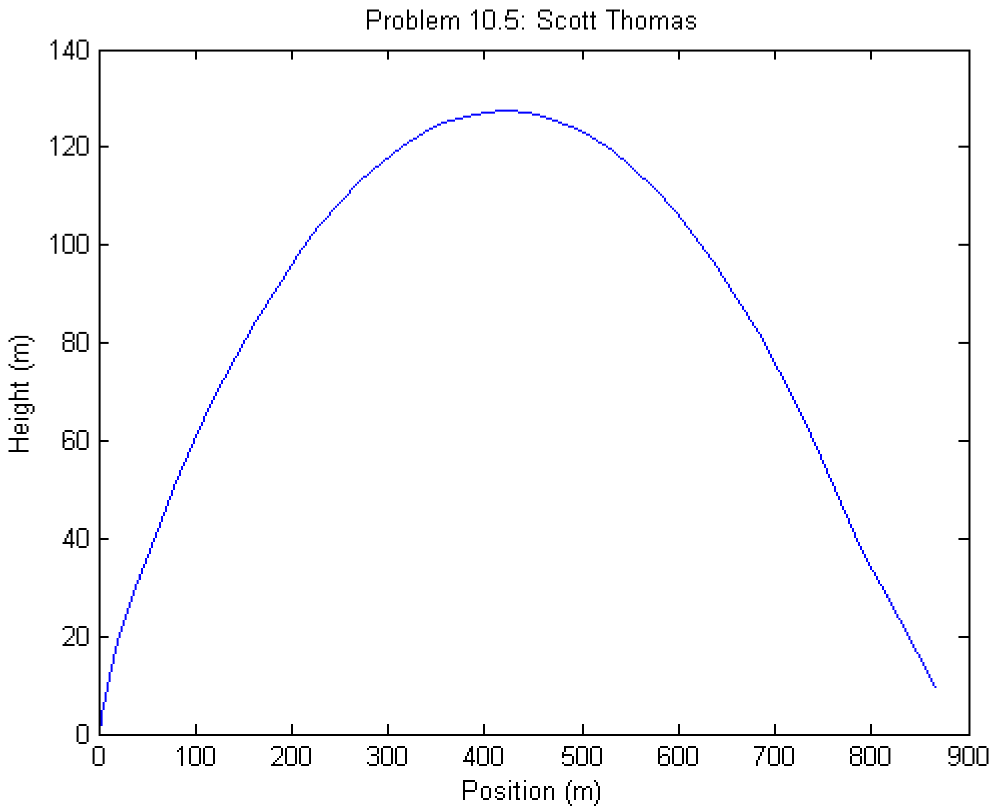
Check results using ode45:

```
function xdot = f10_5x( t,x )

xdot(1) = x(2);
xdot(2) = 0;
xdot = [xdot(1); xdot(2)];
end
```

```
function ydot = f10_5y( t,y )

g = 9.81;
ydot(1) = y(2);
ydot(2) = -g;
ydot = [ydot(1); ydot(2)];
end
```

```
% Problem 10.5: Solve using ode45
clear
clc
disp('Problem 10.5: Scott Thomas')

[t,x] = ode45(@f10_5x, [0, 10], [0, 100*cos(30*pi/180)] );
[t,y] = ode45(@f10_5y, [0, 10], [0, 100*sin(30*pi/180)] );
plot(x(:,1), y(:,1))
ylabel('Height (m)')
xlabel('Position (m)')
title('Problem 10.5: Scott Thomas')
```

```
Problem 10.5: Scott Thomas
```


Problem 10.5: Scott Thomas
Height (m)
Position (m)
0
20
40
60
80
100
120
140
0
100
200
300
400
500
600
700
800
900

Problem 10.8:

8. A tank having vertical sides and a bottom area of 100 ft^2 is used to store water. To fill the tank, water is pumped into the top at the rate given in the following table. Use Simulink to solve for and plot the water height $h(t)$ for $0 \leq t \leq 10$ min.

Time (min)	0	1	2	3	4	5	6	7	8	9	10
Flow Rate (ft^3/min)	0	80	130	150	150	160	165	170	160	140	120

Volume as a function of tank height:

$$V = hA$$

Take the derivative of both sides (A = constant):

$$\frac{dV}{dt} = \frac{dh}{dt} A$$

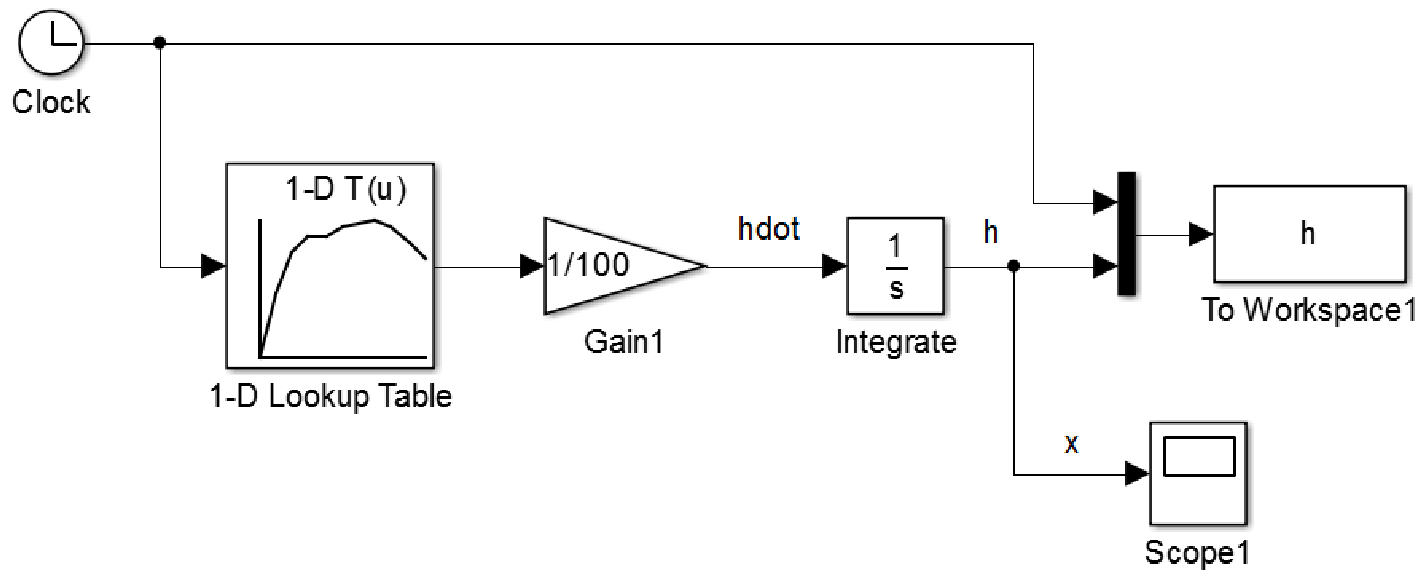
Solve for the rate of change of the height of liquid in the tank:

$$\frac{dh}{dt} = \left(\frac{1}{A}\right) \frac{dV}{dt}$$

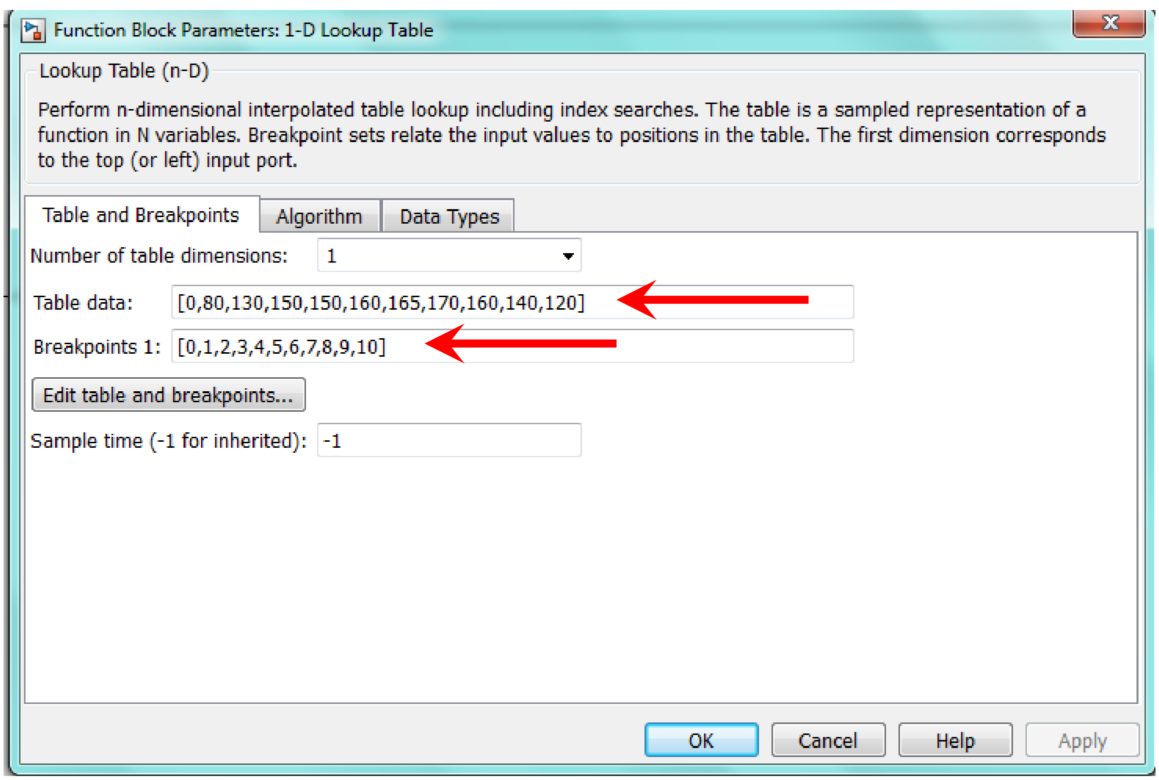
$$\dot{h} = \left(\frac{1}{A}\right) \dot{V}$$

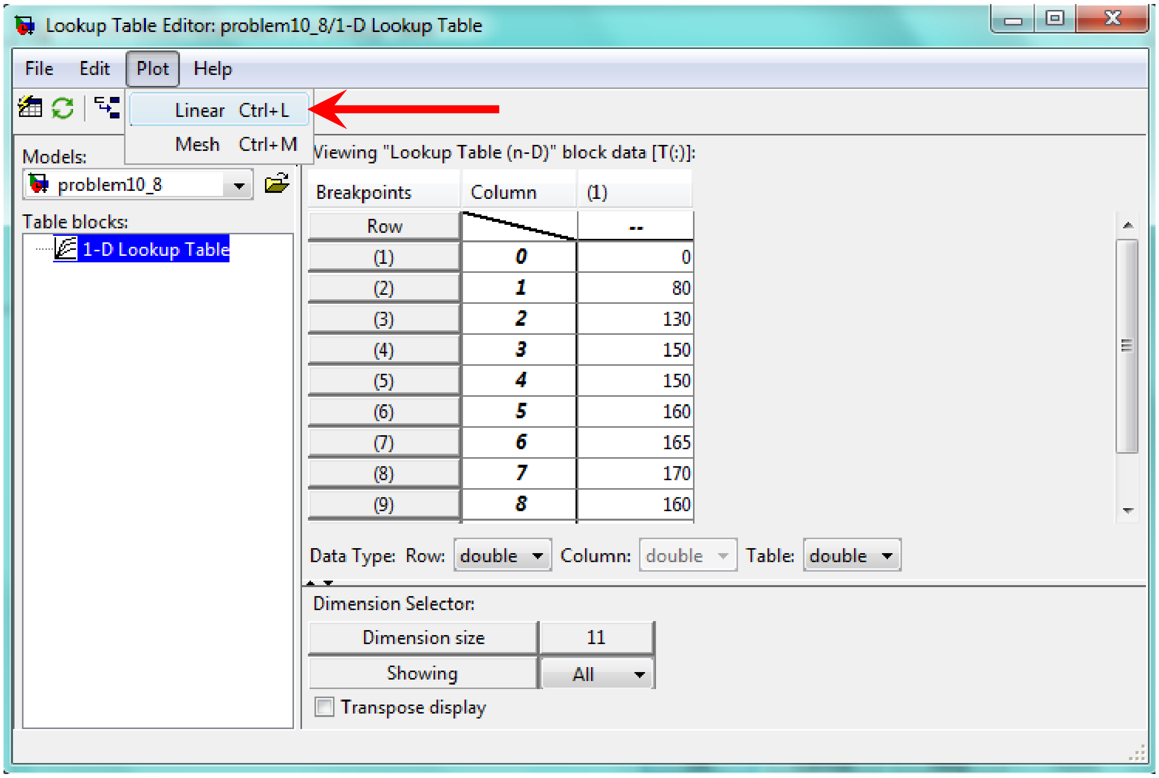
where $\dot{V}$ is the volumetric flow rate of liquid into the tank.

Simulink Model:

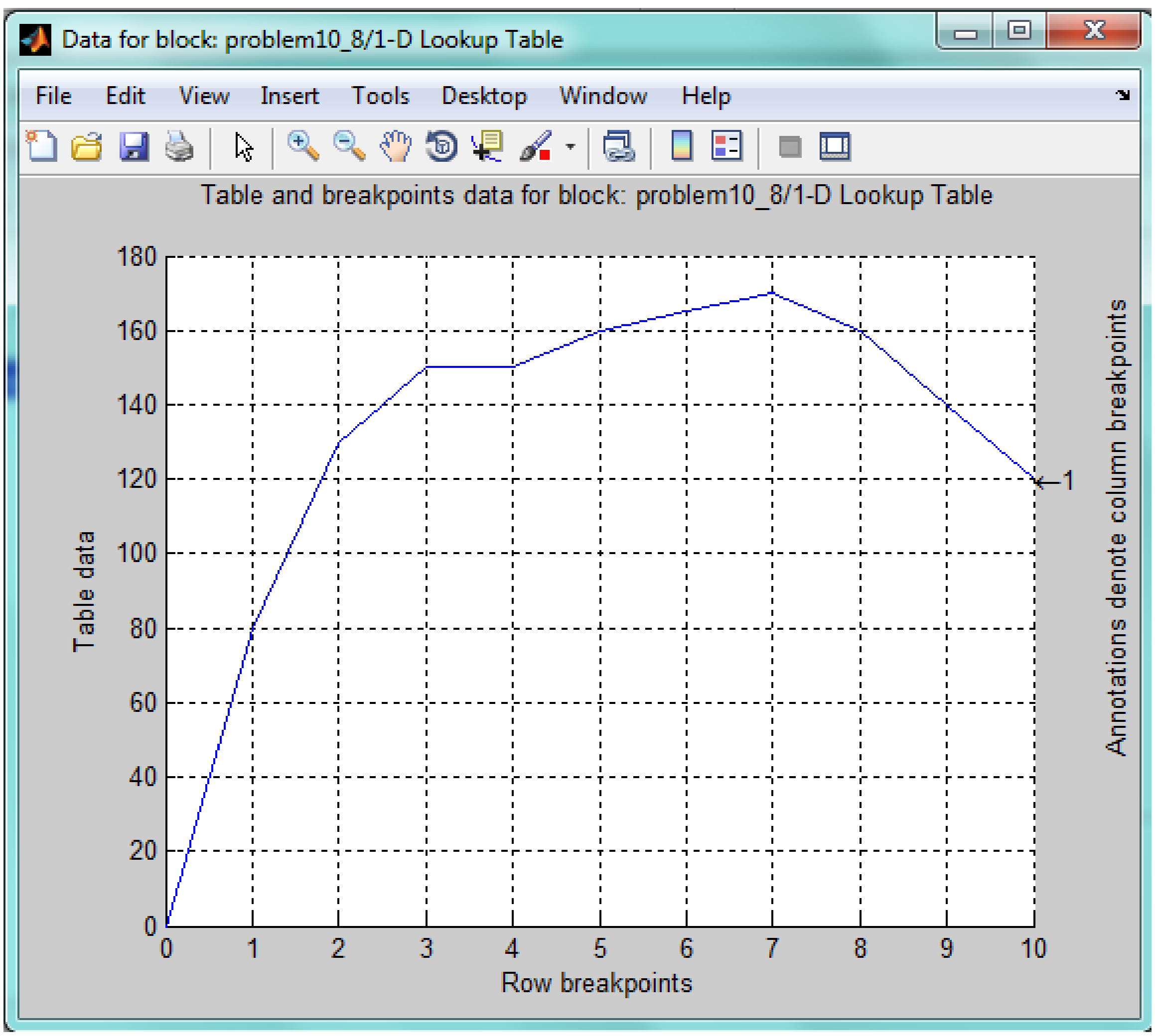


1-D Lookup Table: Linear Interpolation between data points.

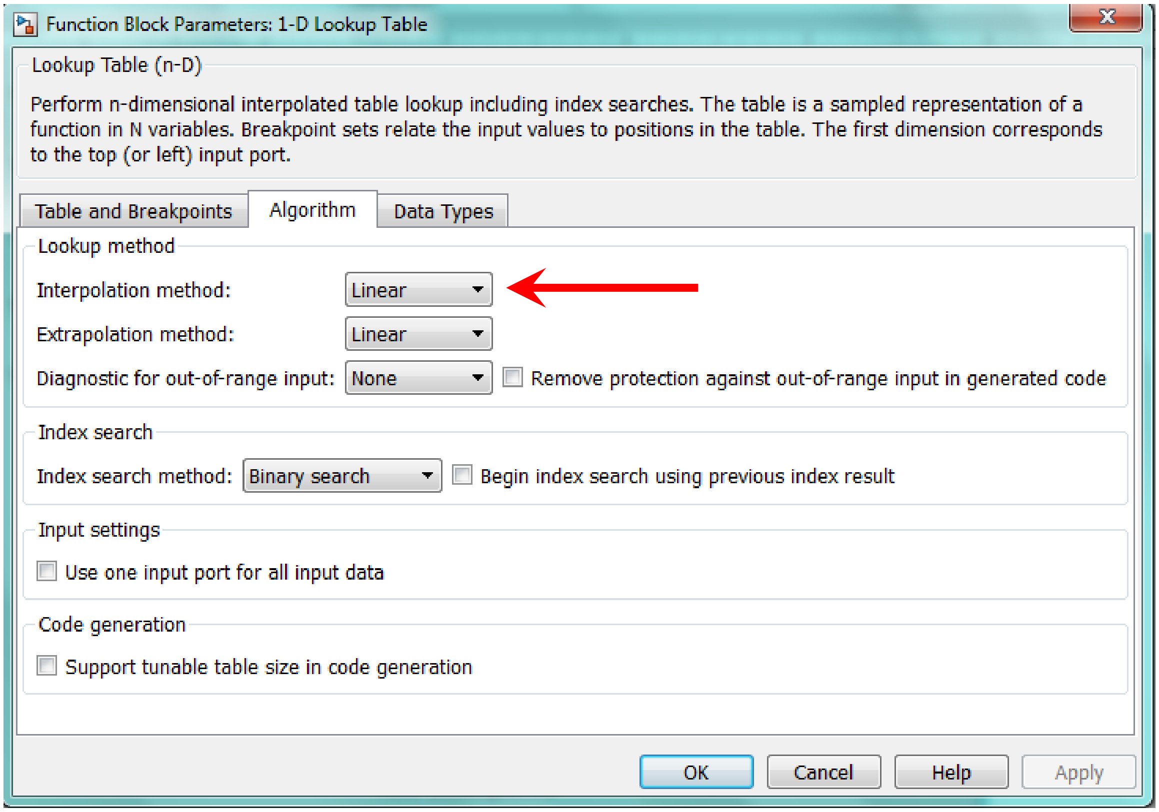


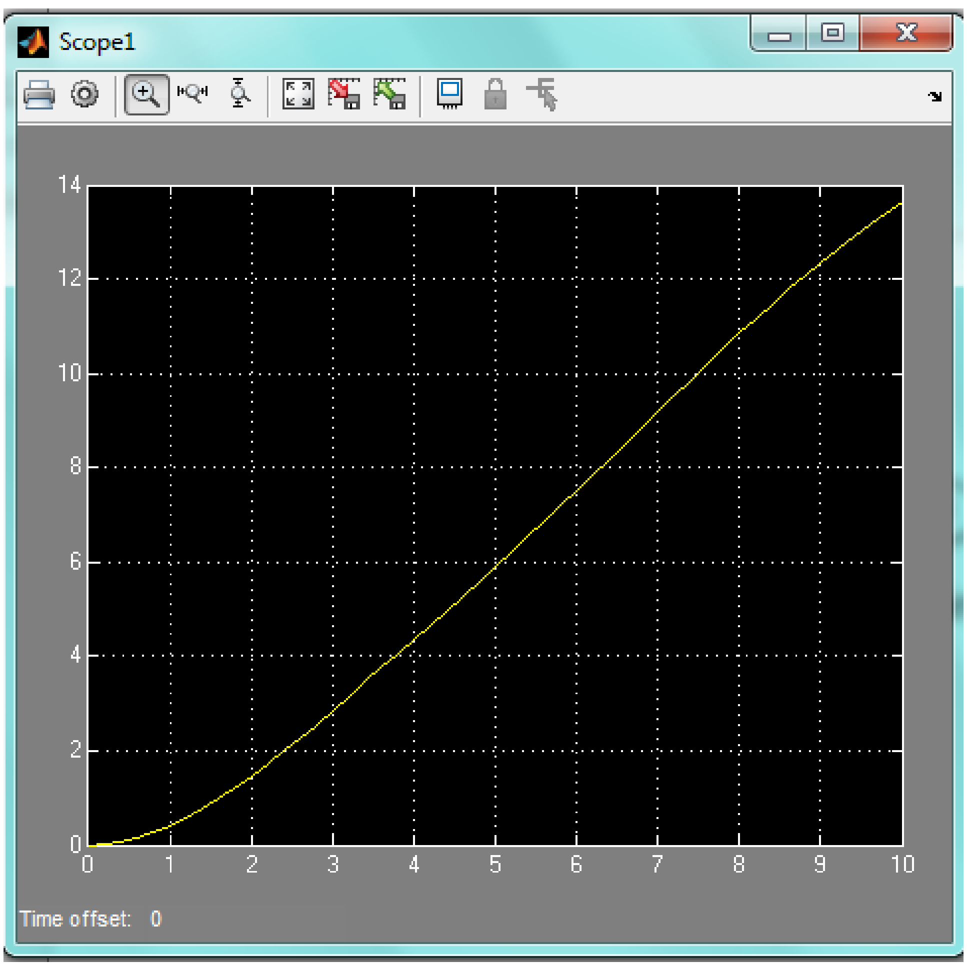


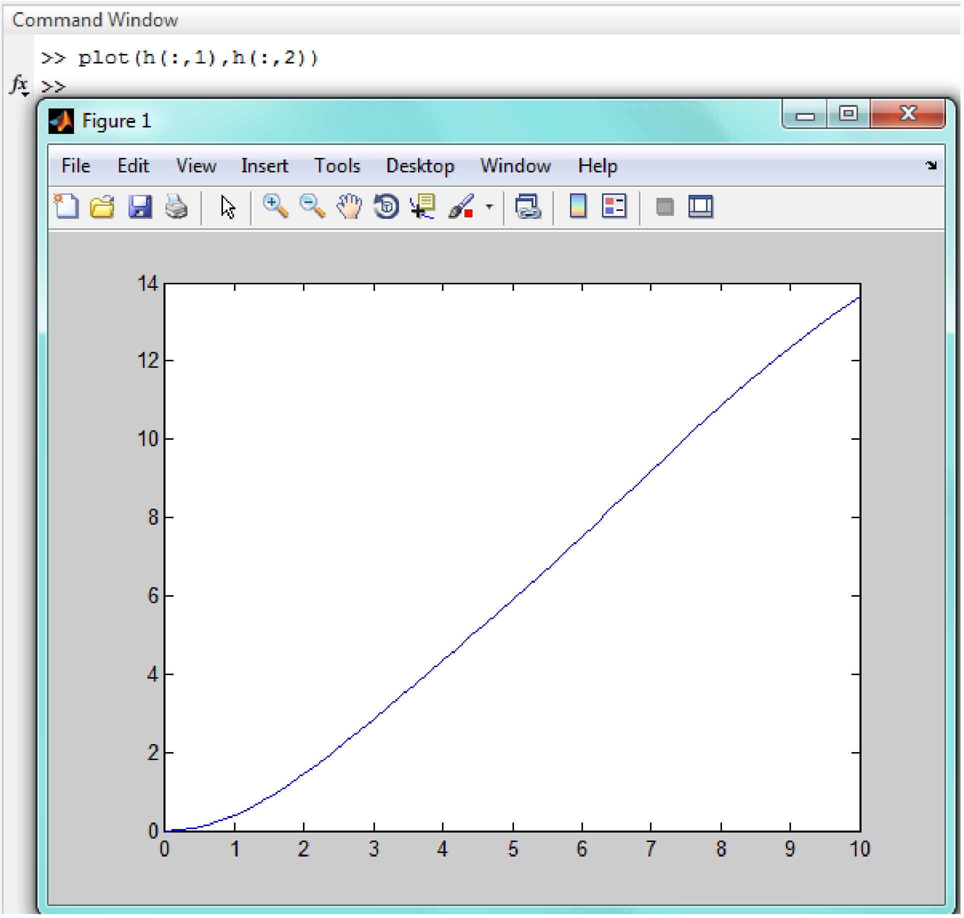
Lookup Table Editor: problem10_8/1-D Lookup Table
File Edit Plot Help
Linear Ctrl+L
Mesh Ctrl+M
Models:
problem10_8
Table blocks:
1-D Lookup Table
Viewing "Lookup Table (n-D)" block data [T(:)]:
Breakpoints Column (1)
Row --
(1) 0 0
(2) 1 80
(3) 2 130
(4) 3 150
(5) 4 150
(6) 5 160
(7) 6 165
(8) 7 170
(9) 8 160
Data Type: Row: double Column: double Table: double
Dimension Selector:
Dimension size 11
Showing All
Transpose display



Data for block: problem10_8/1-D Lookup Table
File Edit View Insert Tools Desktop Window Help
Table and breakpoints data for block: problem10_8/1-D Lookup Table
Table data
Row breakpoints
Annotations denote column breakpoints
1


Function Block Parameters: 1-D Lookup Table
Lookup Table (n-D)
Perform n-dimensional interpolated table lookup including index searches. The table is a sampled representation of a function in N variables. Breakpoint sets relate the input values to positions in the table. The first dimension corresponds to the top (or left) input port.
Table and Breakpoints
Algorithm
Data Types
Lookup method
Interpolation method:
Linear
Extrapolation method:
Linear
Diagnostic for out-of-range input:
None
Remove protection against out-of-range input in generated code
Index search
Index search method:
Binary search
Begin index search using previous index result
Input settings
Use one input port for all input data
Code generation
Support tunable table size in code generation
OK
Cancel
Help
Apply


Scope1
14
12
10
8
6
4
2
0
0
1
2
3
4
5
6
7
8
9
10
Time offset: 0

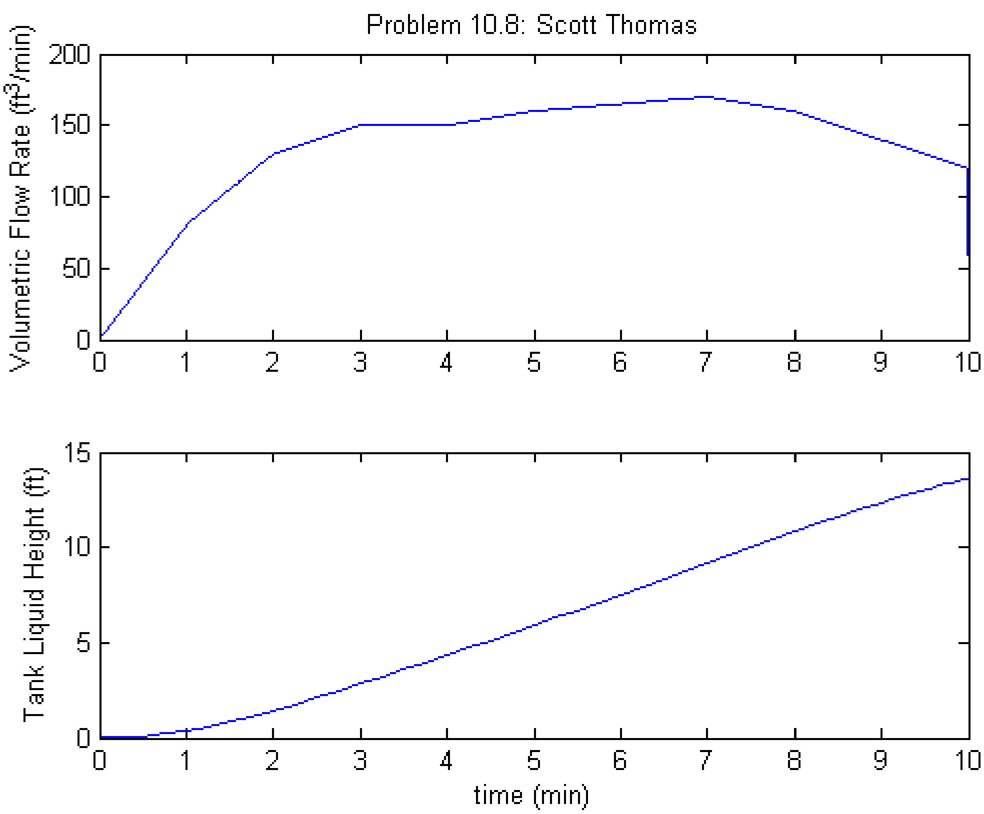

Command Window
>> plot(h(:,1),h(:,2))
>>
Figure 1
File Edit View Insert Tools Desktop Window Help
14
12
10
8
6
4
2
0
0 1 2 3 4 5 6 7 8 9 10

```
function hdot = f10_8( t,~ )

Vdot_data = [0 80 130 150 150 160 165 170 160 140 120];
t_data = [0:10];

Vdot = 0;

if     t >=0 & t < 1
    Vdot = Vdot_data(1) + (Vdot_data(2) - Vdot_data(1))*(t-t_data(1));
elseif t >=1 & t < 2
    Vdot = Vdot_data(2) + (Vdot_data(3) - Vdot_data(2))*(t-t_data(2));
elseif t >=2 & t < 3
    Vdot = Vdot_data(3) + (Vdot_data(4) - Vdot_data(3))*(t-t_data(3));
elseif t >=3 & t < 4
    Vdot = Vdot_data(4) + (Vdot_data(5) - Vdot_data(4))*(t-t_data(4));
elseif t >=4 & t < 5
    Vdot = Vdot_data(5) + (Vdot_data(6) - Vdot_data(5))*(t-t_data(5));
elseif t >=5 & t < 6
    Vdot = Vdot_data(6) + (Vdot_data(7) - Vdot_data(6))*(t-t_data(6));
elseif t >=6 & t < 7
    Vdot = Vdot_data(7) + (Vdot_data(8) - Vdot_data(7))*(t-t_data(7));
elseif t >=7 & t < 8
    Vdot = Vdot_data(8) + (Vdot_data(9) - Vdot_data(8))*(t-t_data(8));
elseif t >=8 & t < 9
    Vdot = Vdot_data(9) + (Vdot_data(10) - Vdot_data(9))*(t-t_data(9));
elseif t >=9 & t < 10
    Vdot = Vdot_data(10) + (Vdot_data(11) - Vdot_data(10))*(t-t_data(10));
end

hdot = (1/100)*Vdot;
end
```

```
% Problem 10.8
clear
clc
disp('Problem 10.8: Scott Thomas')

[th,h] = ode45(@f10_8, [0, 10], 0 );

Vdot_data = [0 80 130 150 150 160 165 170 160 140 120];
t_data = [0:10];

N = 1000;
t = linspace(0,10,N);
Vdot = zeros(1,N);
for k = 1:N

if     t(k) >=0 & t(k) < 1
    Vdot(k) = Vdot_data(1) + (Vdot_data(2) - Vdot_data(1))*(t(k)-t_data(1));
elseif t(k) >=1 & t(k) < 2
    Vdot(k) = Vdot_data(2) + (Vdot_data(3) - Vdot_data(2))*(t(k)-t_data(2));
elseif t(k) >=2 & t(k) < 3
    Vdot(k) = Vdot_data(3) + (Vdot_data(4) - Vdot_data(3))*(t(k)-t_data(3));
elseif t(k) >=3 & t(k) < 4
    Vdot(k) = Vdot_data(4) + (Vdot_data(5) - Vdot_data(4))*(t(k)-t_data(4));
elseif t(k) >=4 & t(k) < 5
```

```
    Vdot(k) = Vdot_data(5) + (Vdot_data(6) - Vdot_data(5))*(t(k)-t_data(5));
elseif t(k) >=5 & t(k) < 6
    Vdot(k) = Vdot_data(6) + (Vdot_data(7) - Vdot_data(6))*(t(k)-t_data(6));
elseif t(k) >=6 & t(k) < 7
    Vdot(k) = Vdot_data(7) + (Vdot_data(8) - Vdot_data(7))*(t(k)-t_data(7));
elseif t(k) >=7 & t(k) < 8
    Vdot(k) = Vdot_data(8) + (Vdot_data(9) - Vdot_data(8))*(t(k)-t_data(8));
elseif t(k) >=8 & t(k) < 9
    Vdot(k) = Vdot_data(9) + (Vdot_data(10) - Vdot_data(9))*(t(k)-t_data(9));
elseif t(k) >=9 & t(k) < 10
    Vdot(k) = Vdot_data(10) + (Vdot_data(11) - Vdot_data(10))*(t(k)-t_data(10));
end
end

subplot(2,1,1)
plot(t,Vdot)%, xlabel('time (min)')
ylabel('Volumetric Flow Rate (ft^3/min)')
title('Problem 10.8: Scott Thomas')

subplot(2,1,2)
plot(th,h), xlabel('time (min)')
ylabel('Tank Liquid Height (ft)')
```

Problem 10.8: Scott Thomas



Problem 10.10:

10. Construct a Simulink model to plot the solution of the following equations for $0 \le t \le 3$

$$\dot{x}_1 = -6x_1 + 4x_2 + f_1(t)$$

$$\dot{x}_2 = 5x_1 - 7x_2 + f_2(t)$$

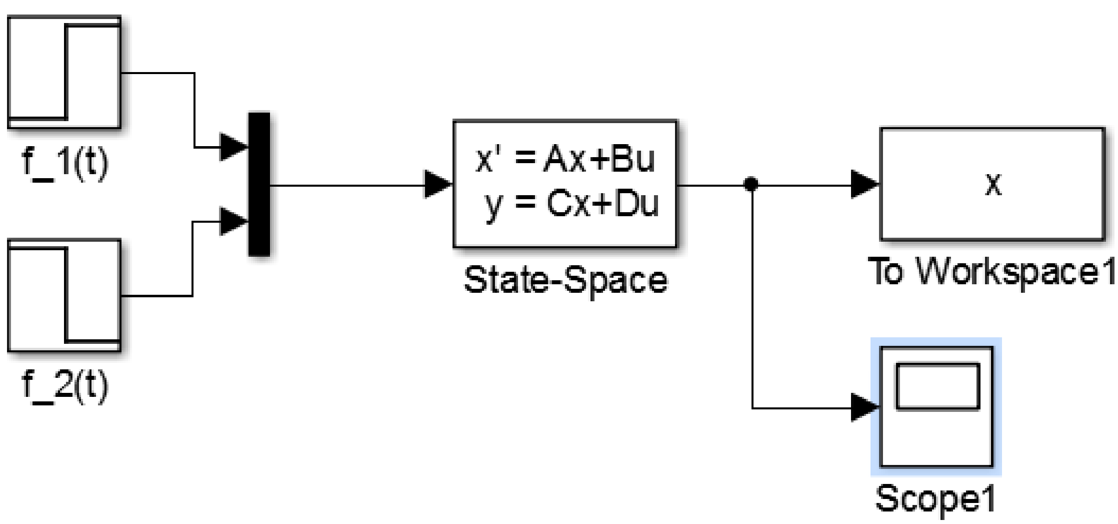
where $f_1(t)$ is a step function of height 3 starting at $t = 0$ and $f_2(t)$ is a step function of height -3 starting at $t = 1$.

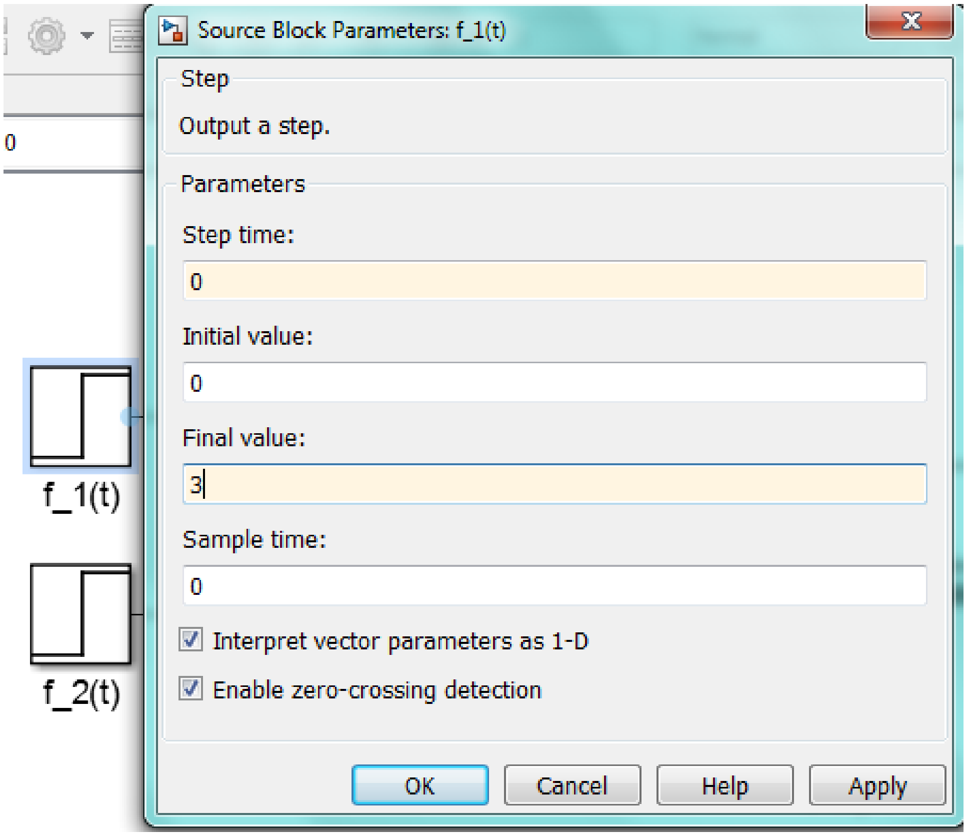
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -6 & 4 \\ 5 & -7 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} f_1(t) \\ f_2(t) \end{bmatrix}$$

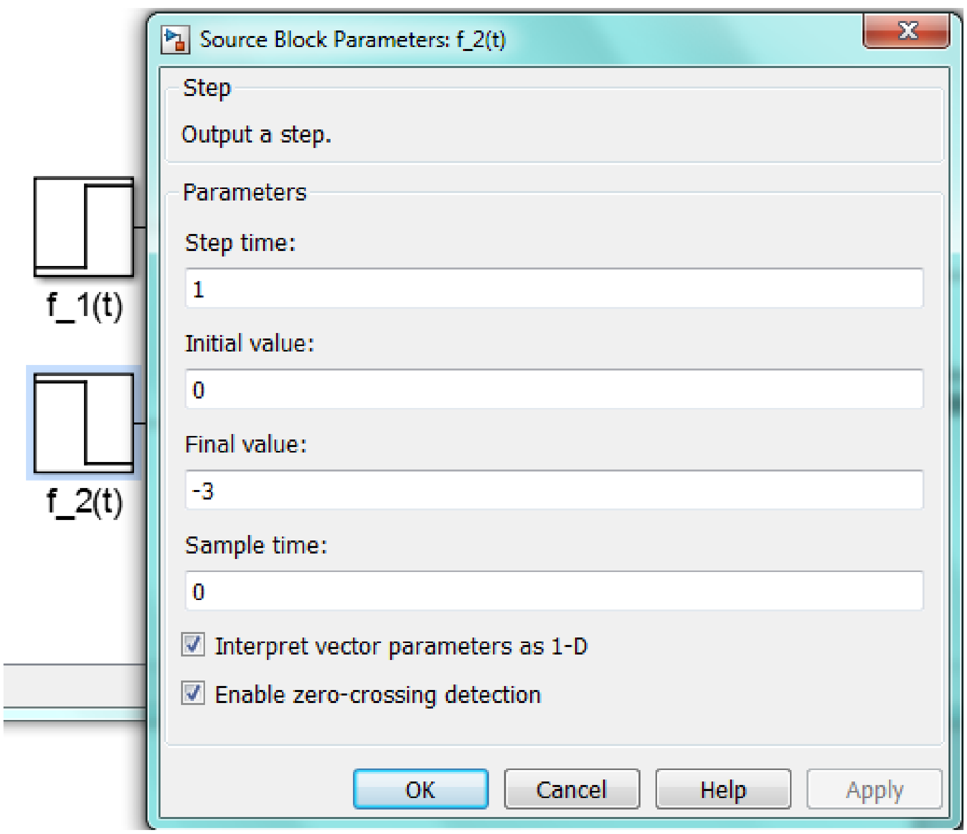
$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u$$

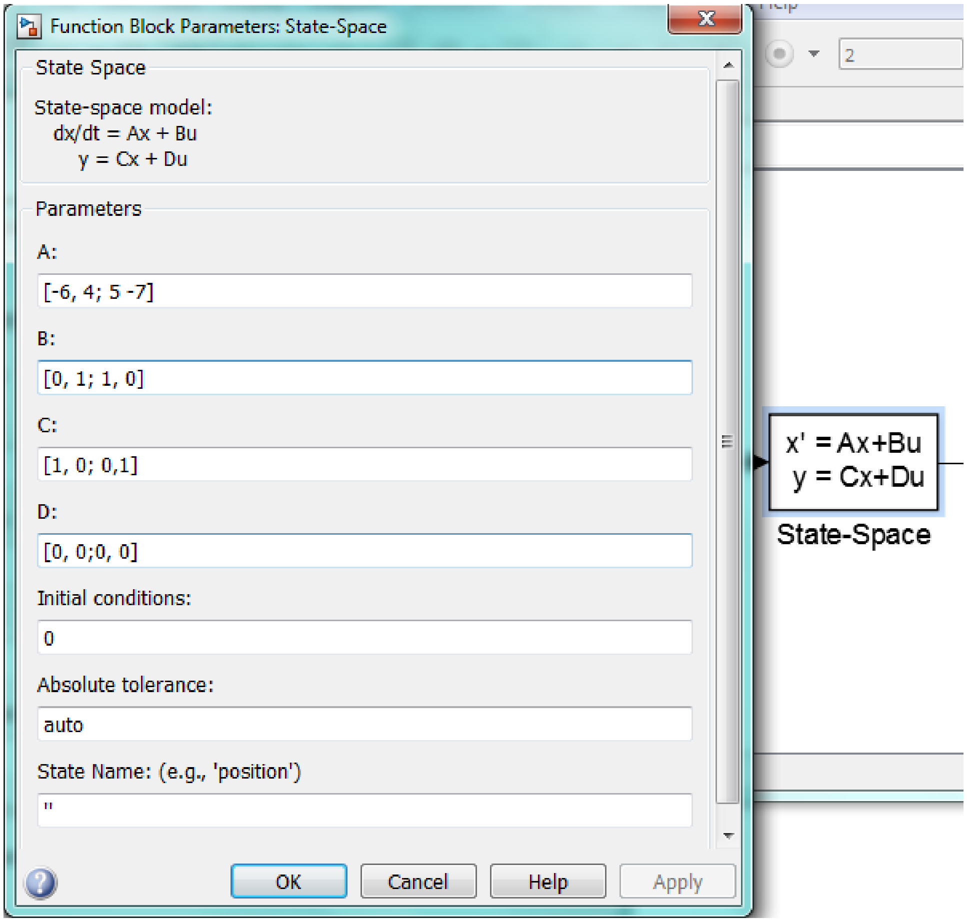
$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_1(t) \\ f_2(t) \end{bmatrix}$$

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}$$

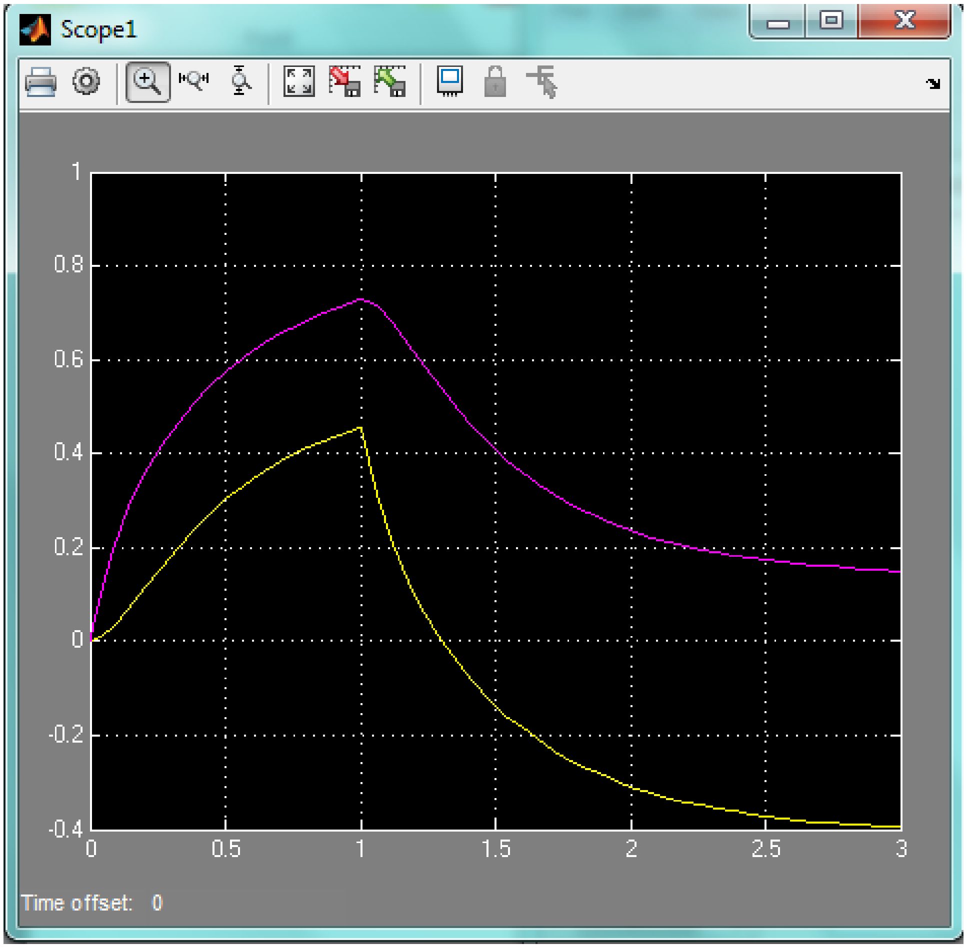



Source Block Parameters: f_1(t)
Step
Output a step.
Parameters
Step time:
0
Initial value:
0
Final value:
3
Sample time:
0
Interpret vector parameters as 1-D
Enable zero-crossing detection
OK
Cancel
Help
Apply
f_1(t)
f_2(t)

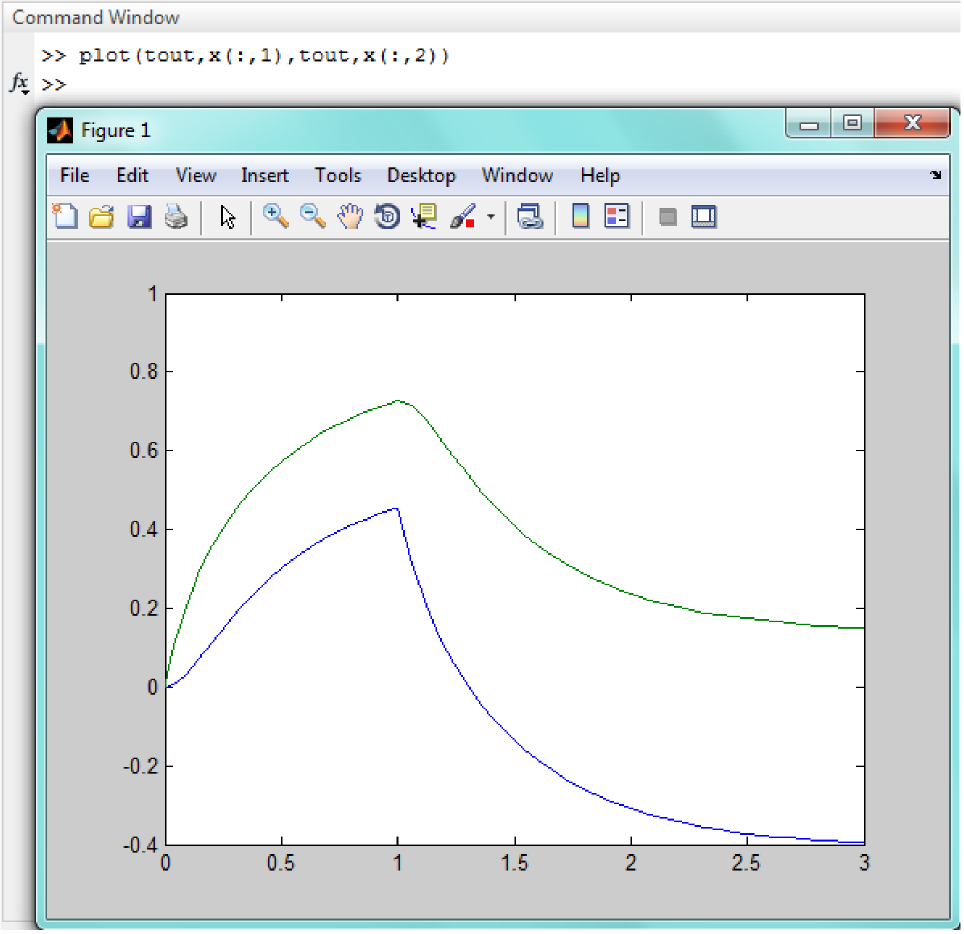

Source Block Parameters: f_2(t)
Step
Output a step.
Parameters
Step time:
1
Initial value:
0
Final value:
-3
Sample time:
0
Interpret vector parameters as 1-D
Enable zero-crossing detection
OK
Cancel
Help
Apply
f_1(t)
f_2(t)



Function Block Parameters: State-Space
State Space
State-space model:
dx/dt = Ax + Bu
y = Cx + Du
Parameters
A:
[-6, 4; 5 -7]
B:
[0, 1; 1, 0]
C:
[1, 0; 0,1]
D:
[0, 0;0, 0]
Initial conditions:
0
Absolute tolerance:
auto
State Name: (e.g., 'position')
''
OK
Cancel
Help
Apply
x' = Ax+Bu
y = Cx+Du
State-Space



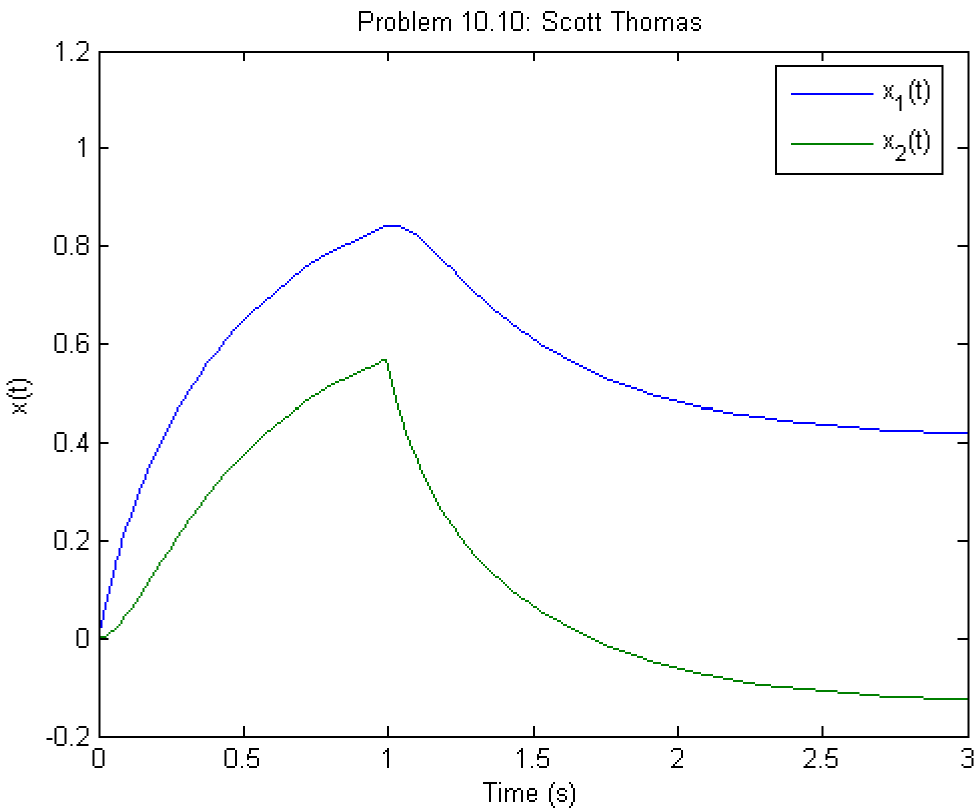
Scope1
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
0.5
1
1.5
2
2.5
3
Time offset: 0




```
function xdot = f10_10( t,x )

f1 = 3;

if t > 1
    f2 = -3;
else
    f2 = 0;
end

xdot(1) = -6*x(1) + 4*x(2) + f1;
xdot(2) = 5*x(1) - 7*x(2) + f2;
xdot = [xdot(1); xdot(2)];
end
```

```
% Problem 10.10: Solve using ode45/matrix method
clear
clc
disp('Problem 10.10: Scott Thomas')

[t,x] = ode45(@f10_10, [0, 3], [0, 0] );
plot(t,x(:,1),t, x(:,2)), xlabel('Time (s)')
ylabel('x(t)')
title('Problem 10.10: Scott Thomas')
legend('x_1(t)', 'x_2(t)','Location', 'Best')
```

Problem 10.10: Scott Thomas



Problem 10.12:

12. Construct a Simulink model of the following problem.

$$5\dot{x} + \sin x = f(t) \qquad x(0) = 0$$

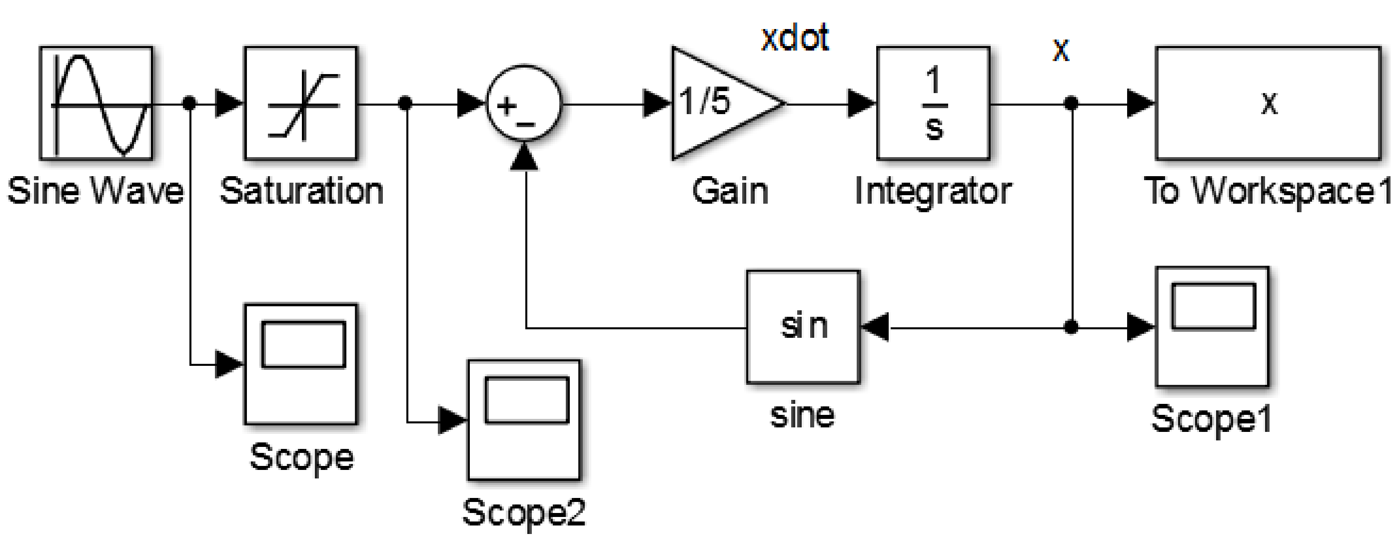
The forcing function is

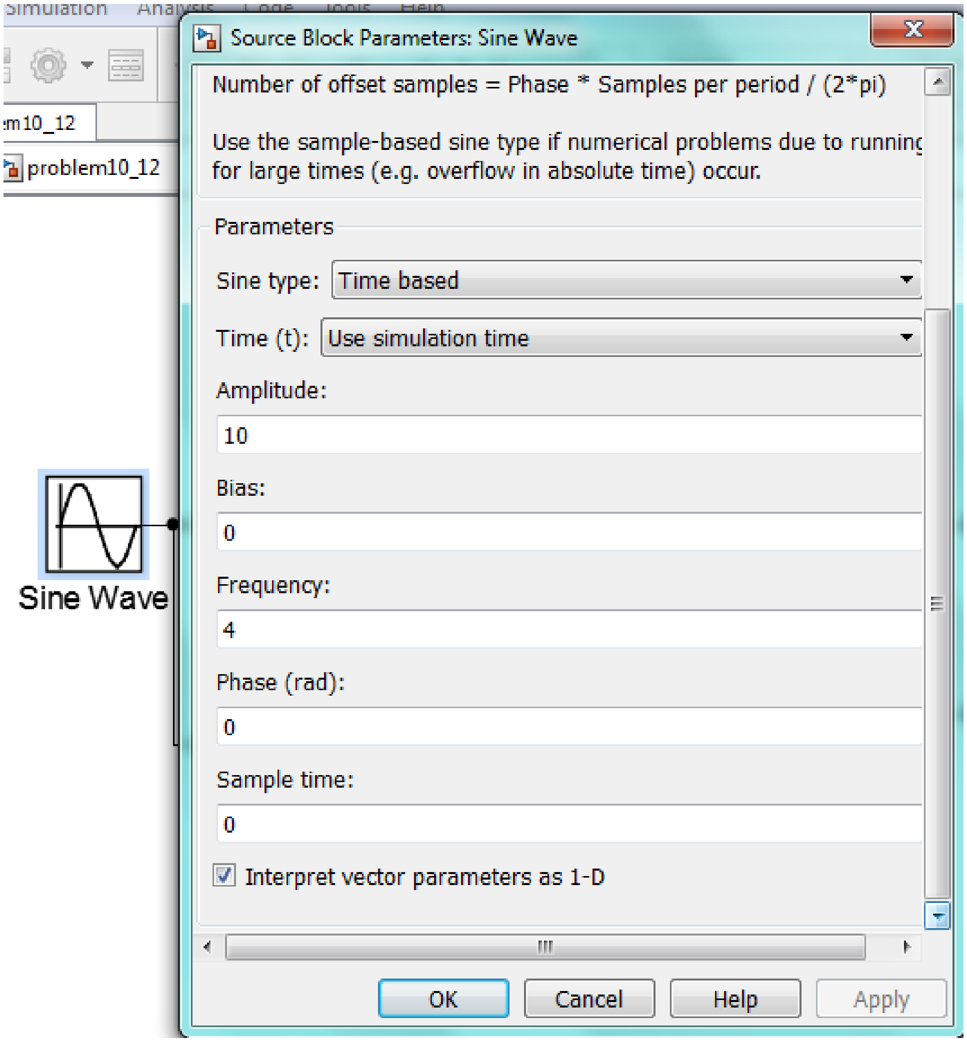
$$f(t) = \begin{cases} -5 & \text{if } g(t) \leq -5 \\ g(t) & \text{if } -5 \leq g(t) \leq 5 \\ 5 & \text{if } g(t) \geq 5 \end{cases}$$

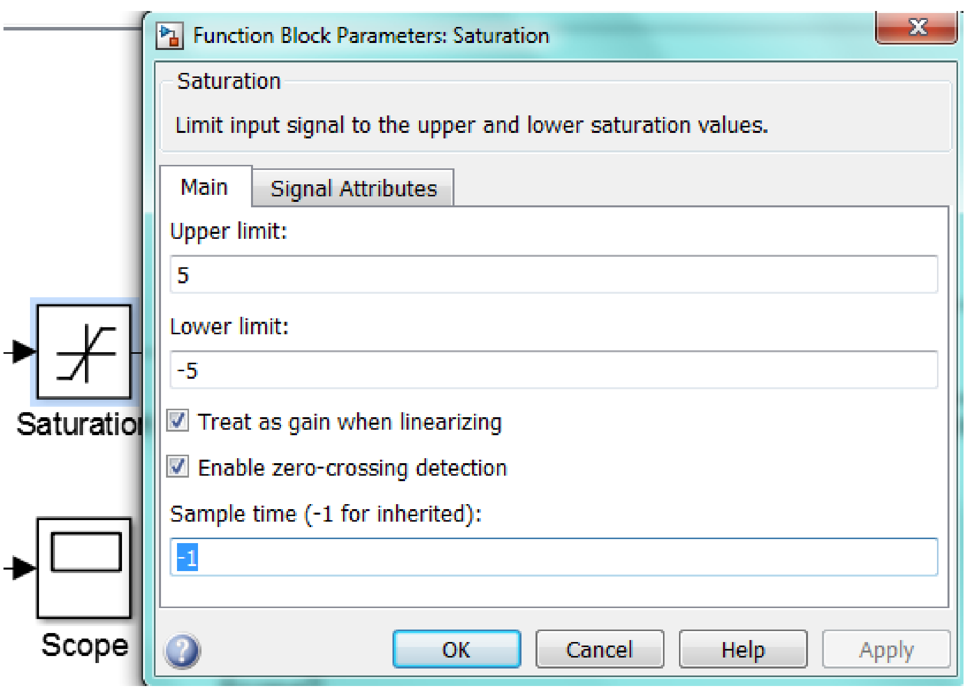
where $g(t) = 10 \sin 4t$.

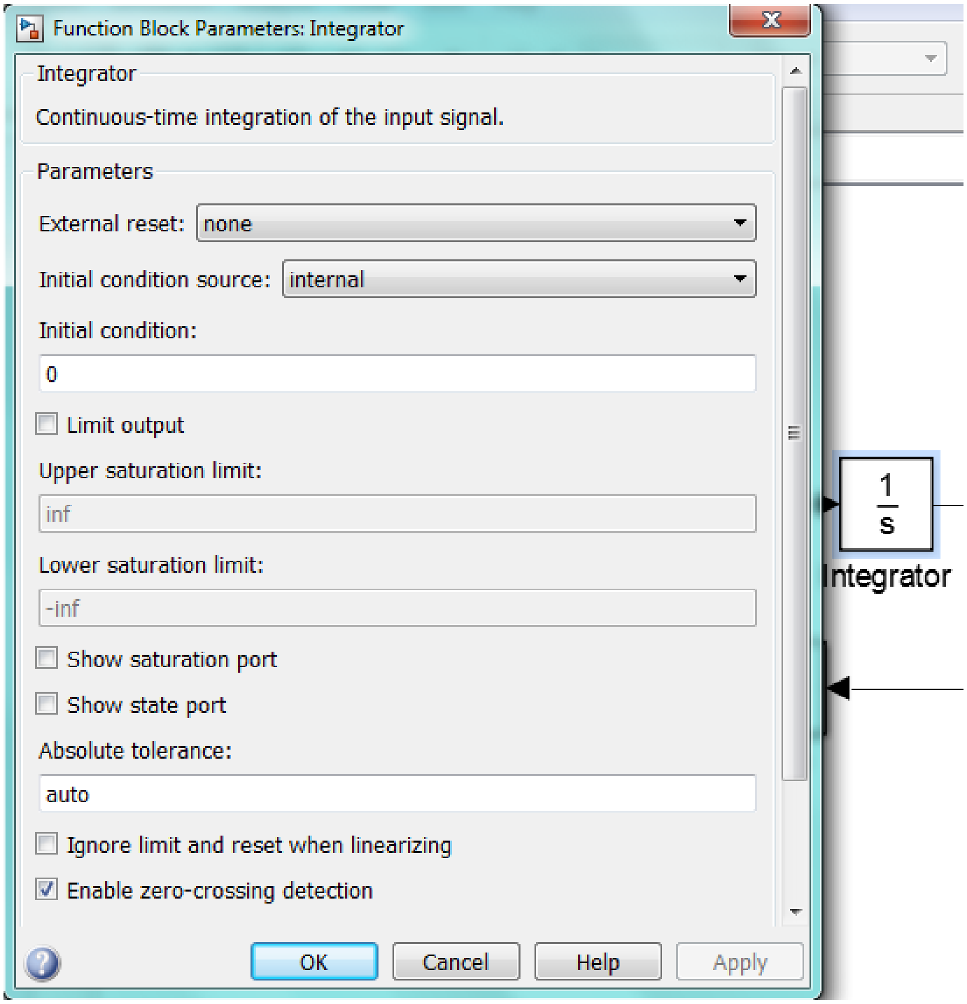
$$\dot{x} = \frac{1}{5}[f(t) - \sin x]$$

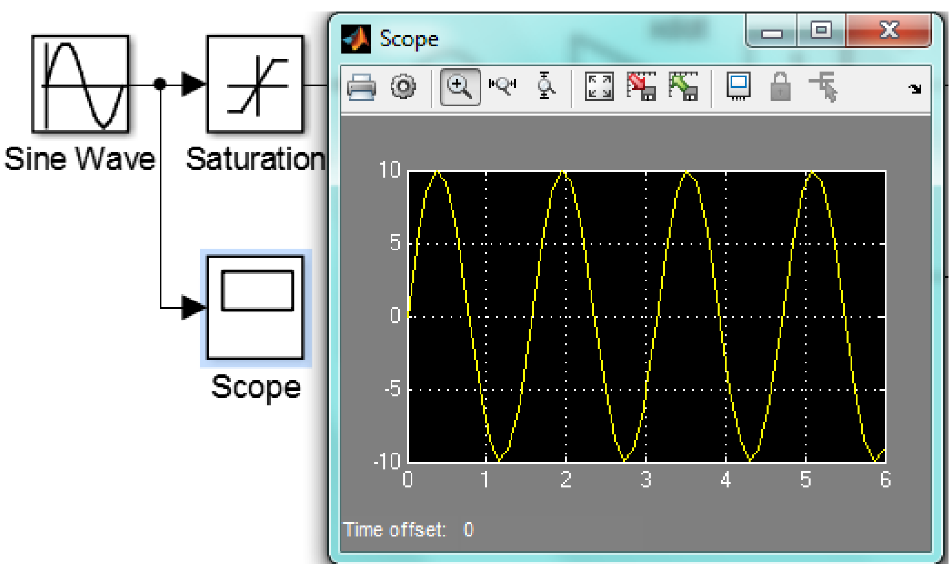
$$x = \int \left\{\frac{1}{5}[f(t) - \sin x]\right\}$$

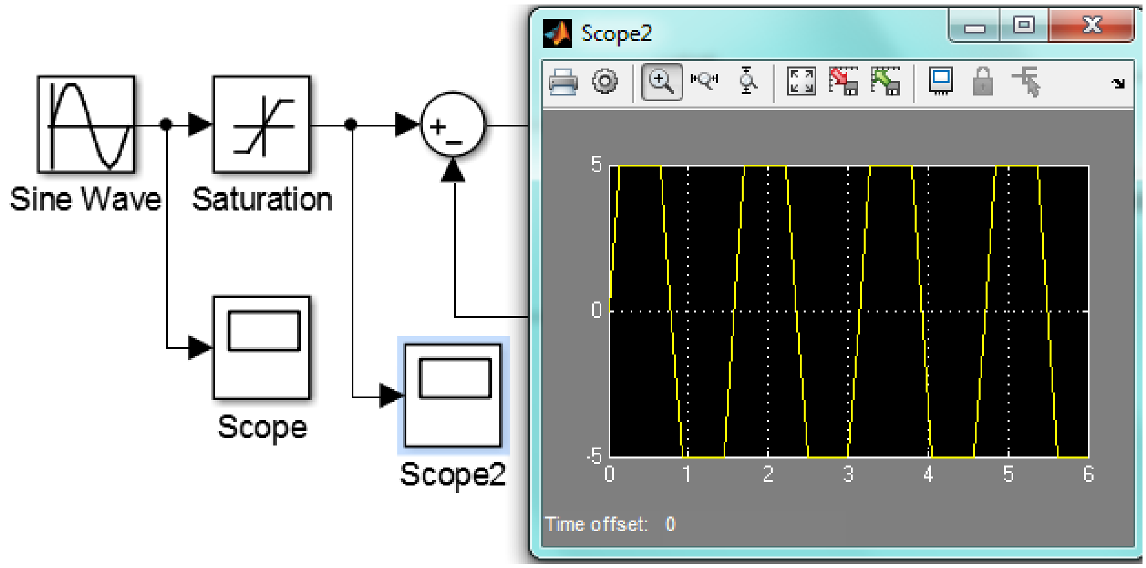


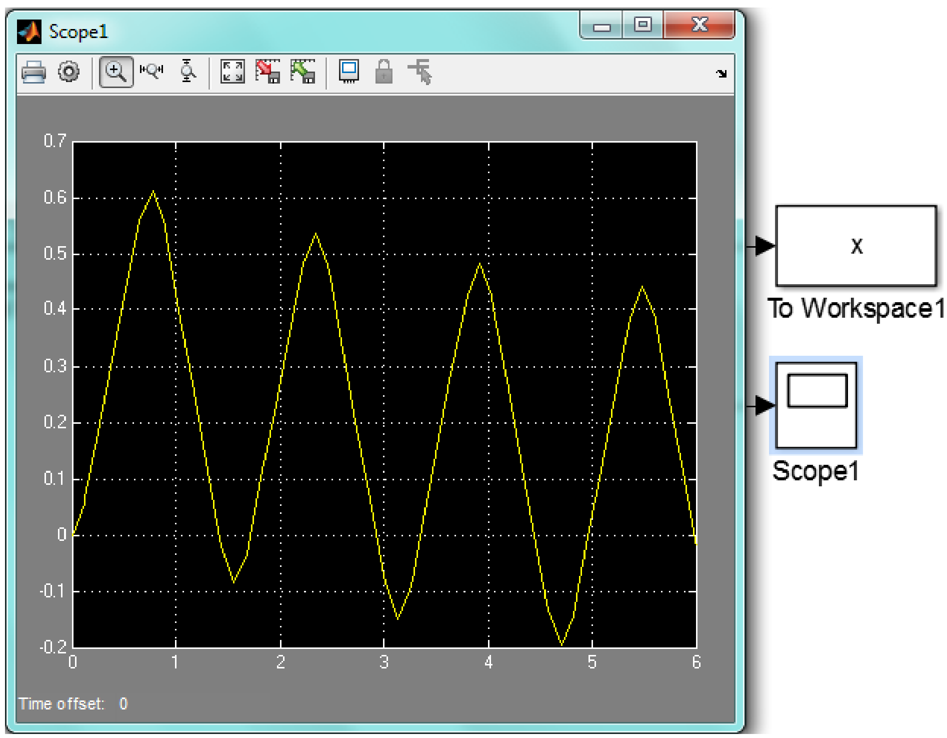

Source Block Parameters: Sine Wave
Number of offset samples = Phase * Samples per period / (2*pi)
Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.
Parameters
Sine type: Time based
Time (t): Use simulation time
Amplitude:
10
Bias:
0
Frequency:
4
Phase (rad):
0
Sample time:
0
Interpret vector parameters as 1-D
OK
Cancel
Help
Apply
Sine Wave
problem10_12


Function Block Parameters: Saturation
Saturation
Limit input signal to the upper and lower saturation values.
Main
Signal Attributes
Upper limit:
5
Lower limit:
-5
Treat as gain when linearizing
Enable zero-crossing detection
Sample time (-1 for inherited):
-1
OK
Cancel
Help
Apply
Scope


Function Block Parameters: Integrator
Integrator
Continuous-time integration of the input signal.
Parameters
External reset: none
Initial condition source: internal
Initial condition:
0
Limit output
Upper saturation limit:
inf
Lower saturation limit:
-inf
Show saturation port
Show state port
Absolute tolerance:
auto
Ignore limit and reset when linearizing
Enable zero-crossing detection
OK
Cancel
Help
Apply
1
s
Integrator

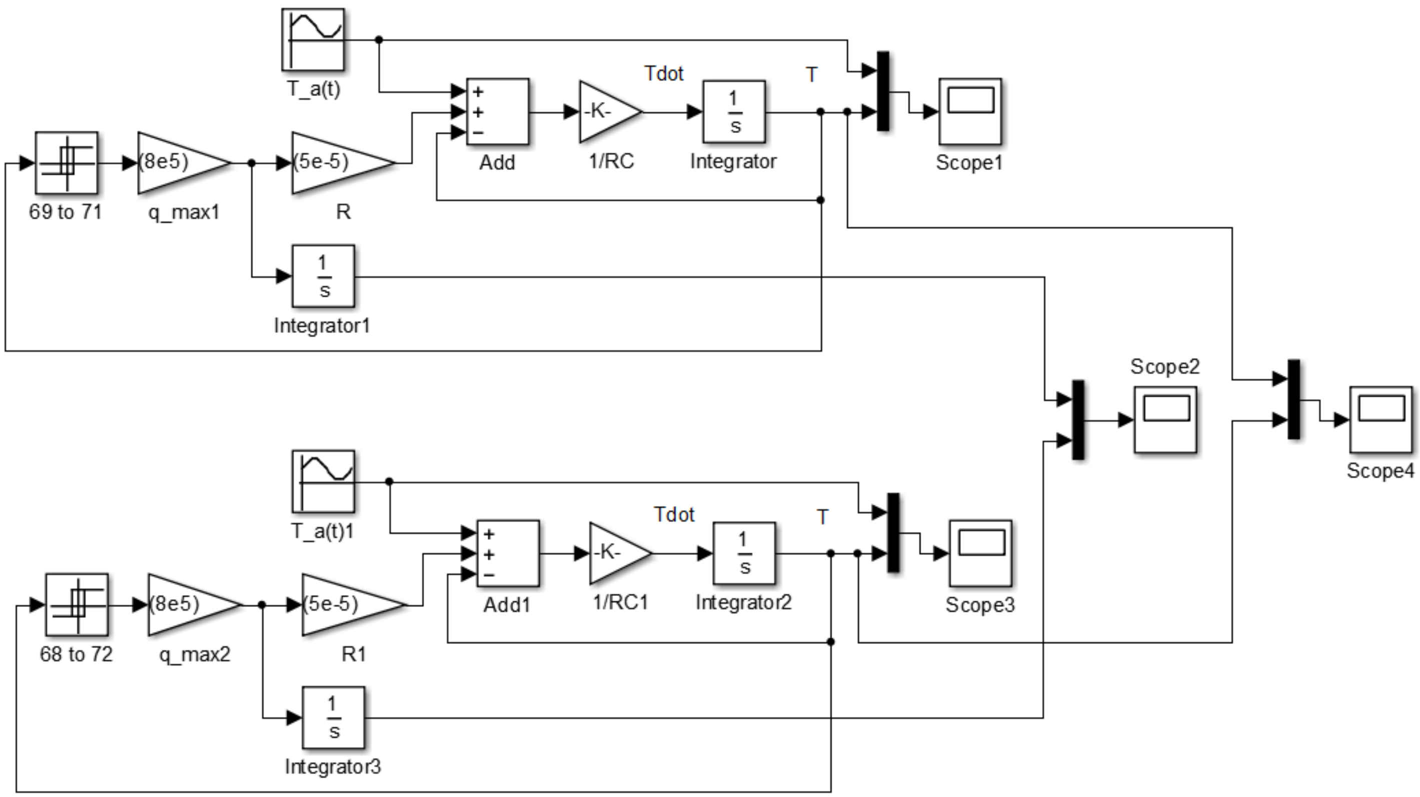

Sine Wave
Saturation
Scope
Scope
10
5
0
-5
-10
0
1
2
3
4
5
6
Time offset: 0


Sine Wave
Saturation
Scope
Scope2
Time offset: 0

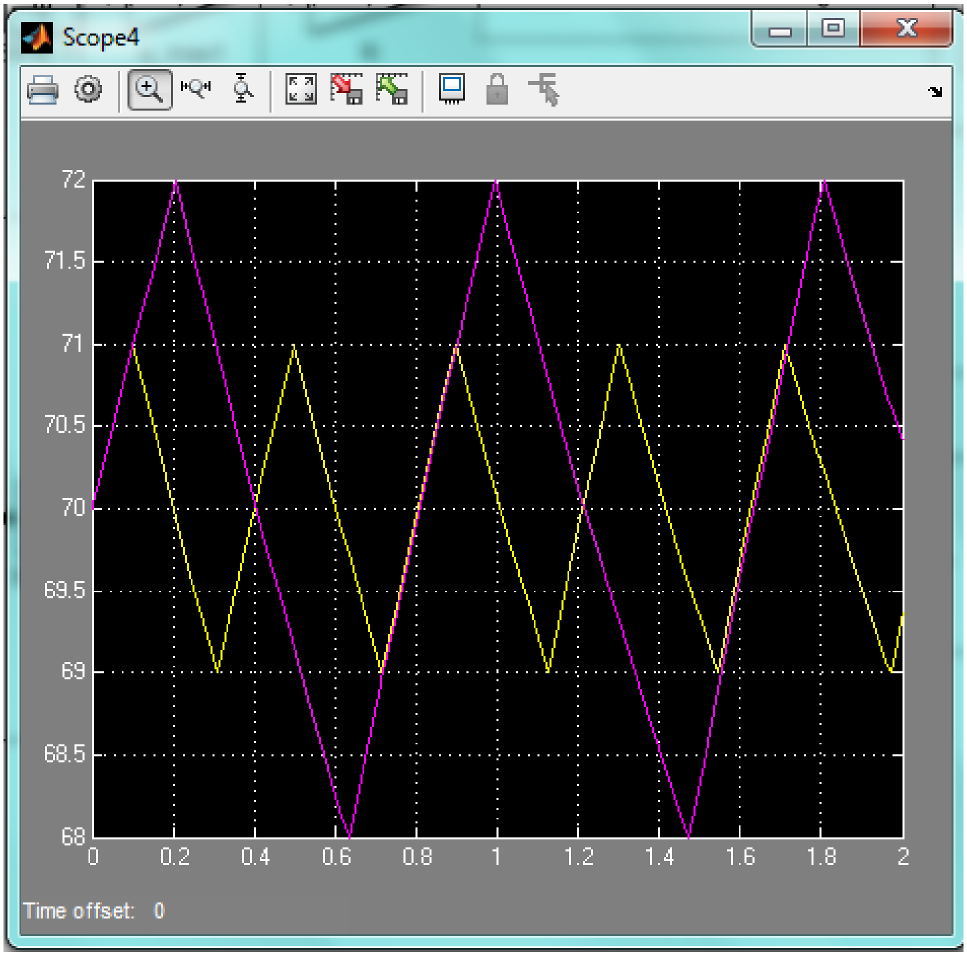

Scope1
Time offset: 0
x
To Workspace1
Scope1

Problem 10.16:

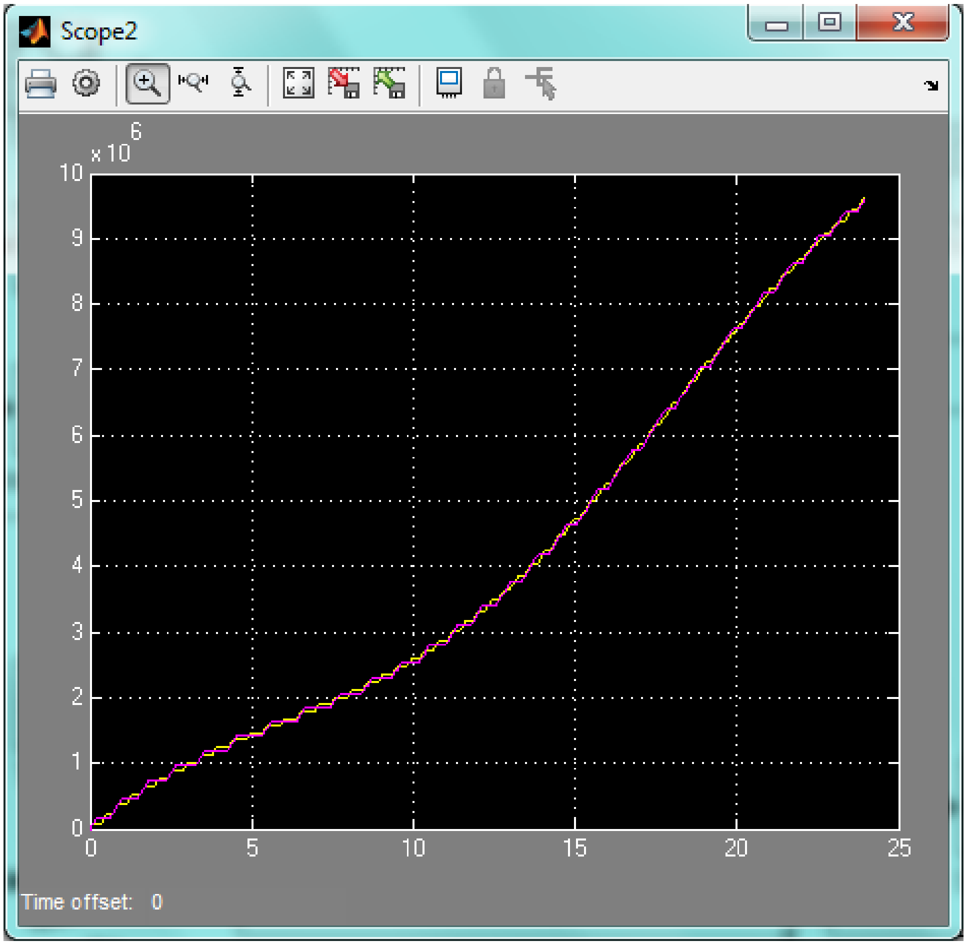
16. Refer to Problem 15. Use the simulation with $q_{max} = 8 \times 10^5$ to compare the energy consumption and the thermostat cycling frequency for the two temperature bands (69°, 71°) and (68°, 72°).



Temperature versus time for 2 hours: This shows the two deadband ranges.



Heat Input over 24 hours: The two traces are nearly identical



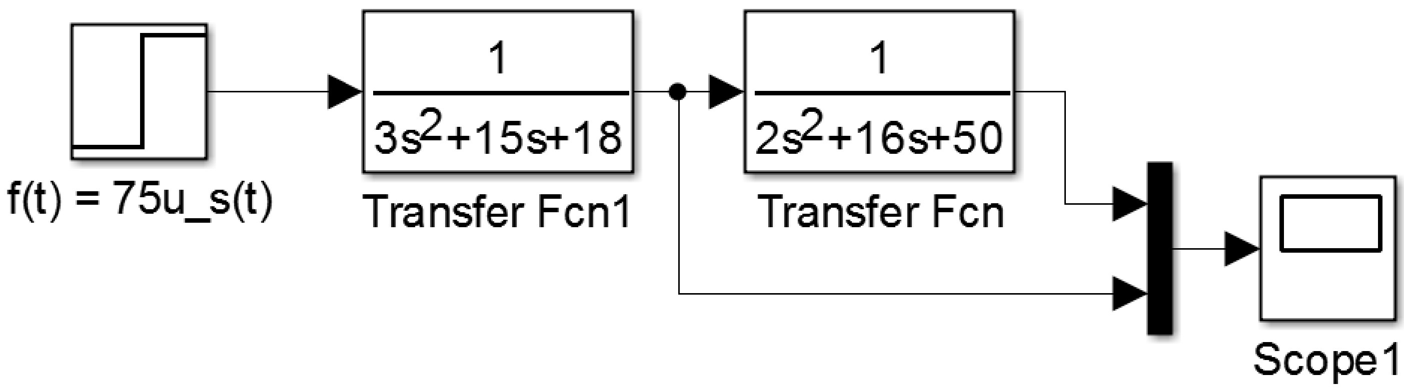
Problem 10.19:

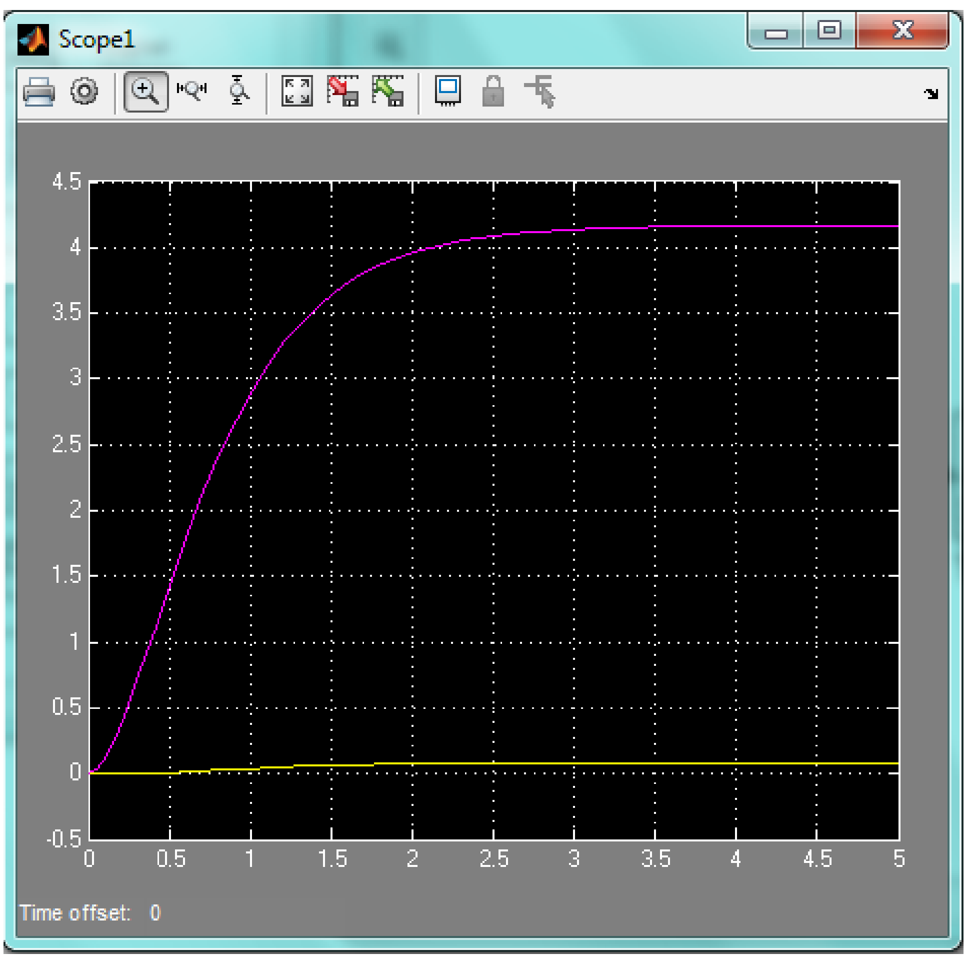
19. Use Transfer Function blocks to construct a Simulink model to plot the solution of the following equations for $0 \le t \le 2$.

$$3\ddot{x} + 15\dot{x} + 18x = f(t) \qquad x(0) = \dot{x}(0) = 0$$

$$2\ddot{y} + 16\dot{y} + 50y = x(t) \qquad y(0) = \dot{y}(0) = 0$$

where $f(t) = 75u_s(t)$.

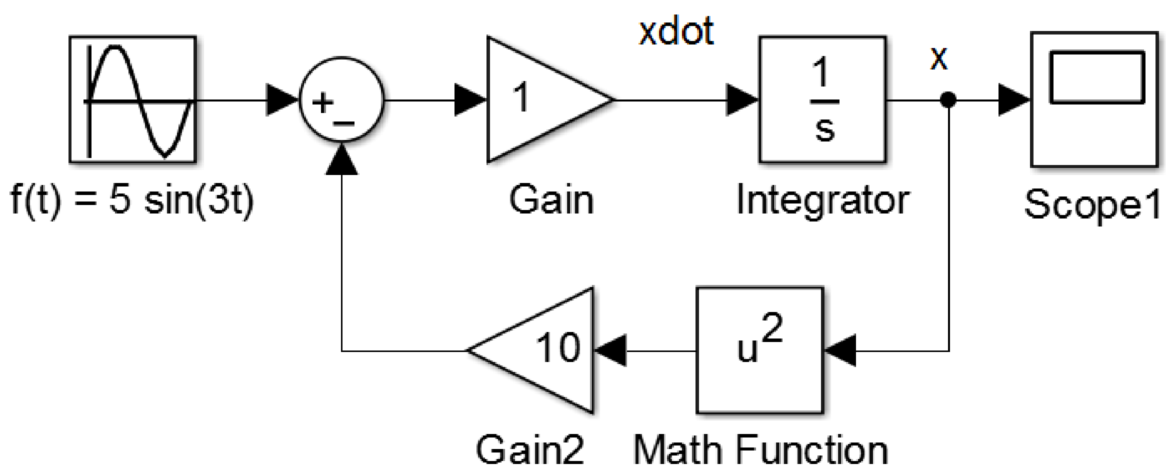


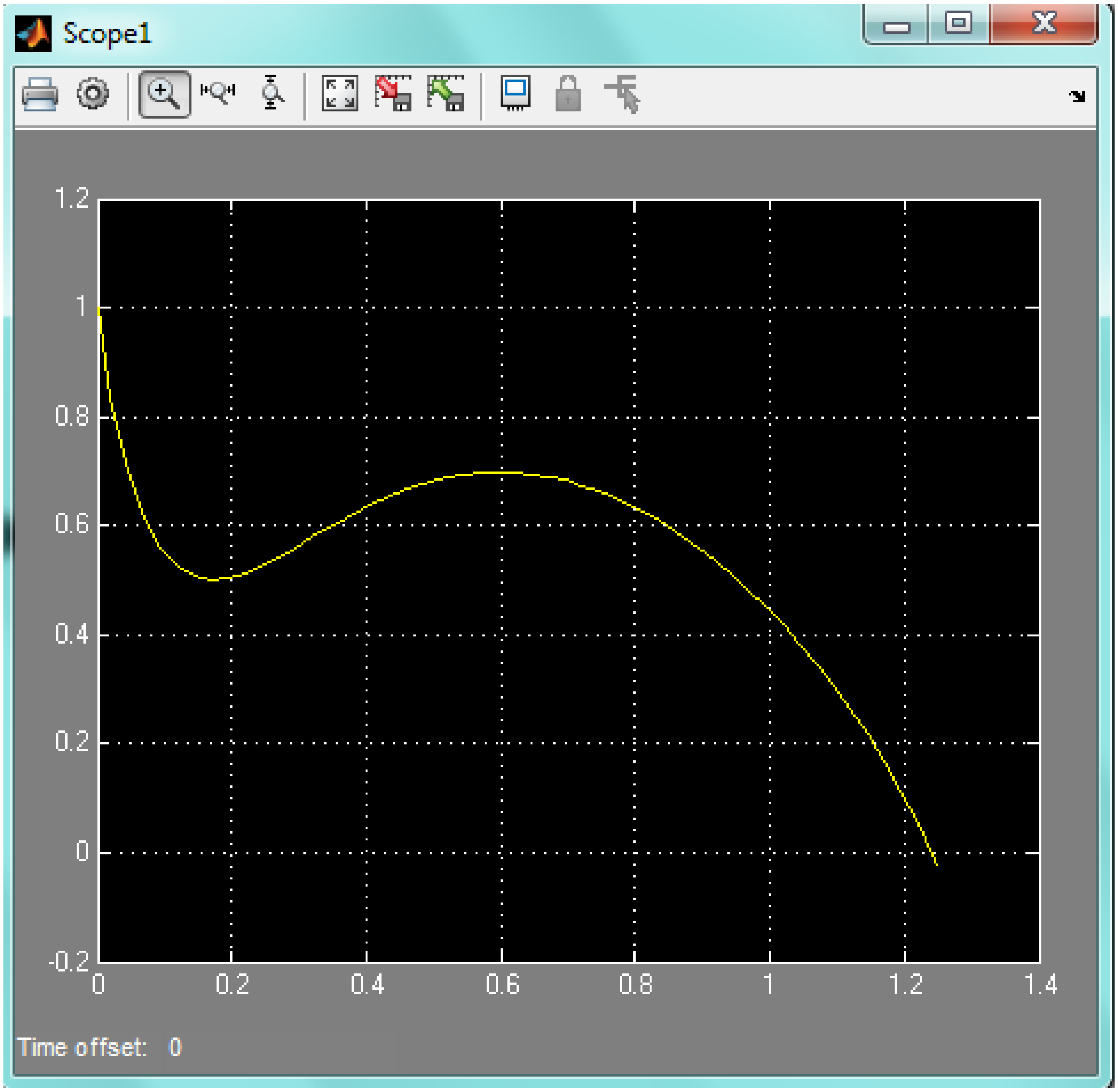


Problem 10.23:

23. Create a Simulink model to plot the solution of the following equation for $0 \le t \le 3$.

$$\dot{x} + 10x^2 = 5 \sin 3t \qquad x(0) = 1$$




Scope1
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
0.2
0.4
0.6
0.8
1
1.2
1.4
Time offset: 0

Problem 10.30:

30. *a.* Use the subsystem block developed in Section 10.7 to construct a Simulink model of the system shown in Figure P30. The mass inflow rate is a step function.

b. Use the Simulink model to obtain plots of $h_1(t)$ and $h_2(t)$ for the following parameter values: $A_1 = 3\ \text{ft}^2$, $A_2 = 5\ \text{ft}^2$, $R_1 = 30\ \text{ft}^{-1} \cdot \text{sec}^{-1}$, $R_2 = 40\ \text{ft}^{-1} \cdot \text{sec}^{-1}$, $\rho = 1.94\ \text{slug/ft}^3$, $q_{mi} = 0.5$ slug/sec, $h_1(0) = 2$ ft, and $h_2(0) = 5$ ft.

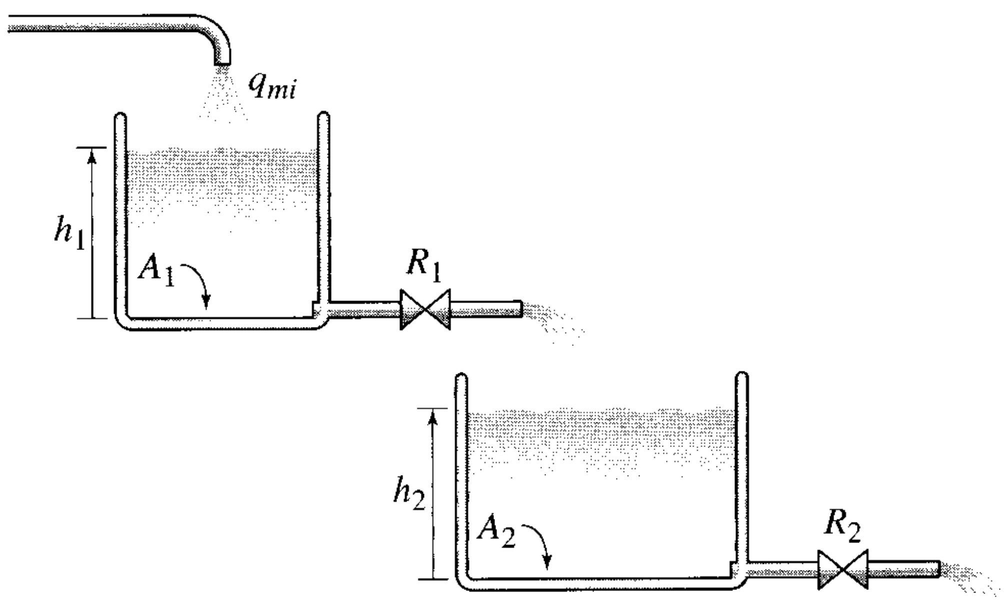
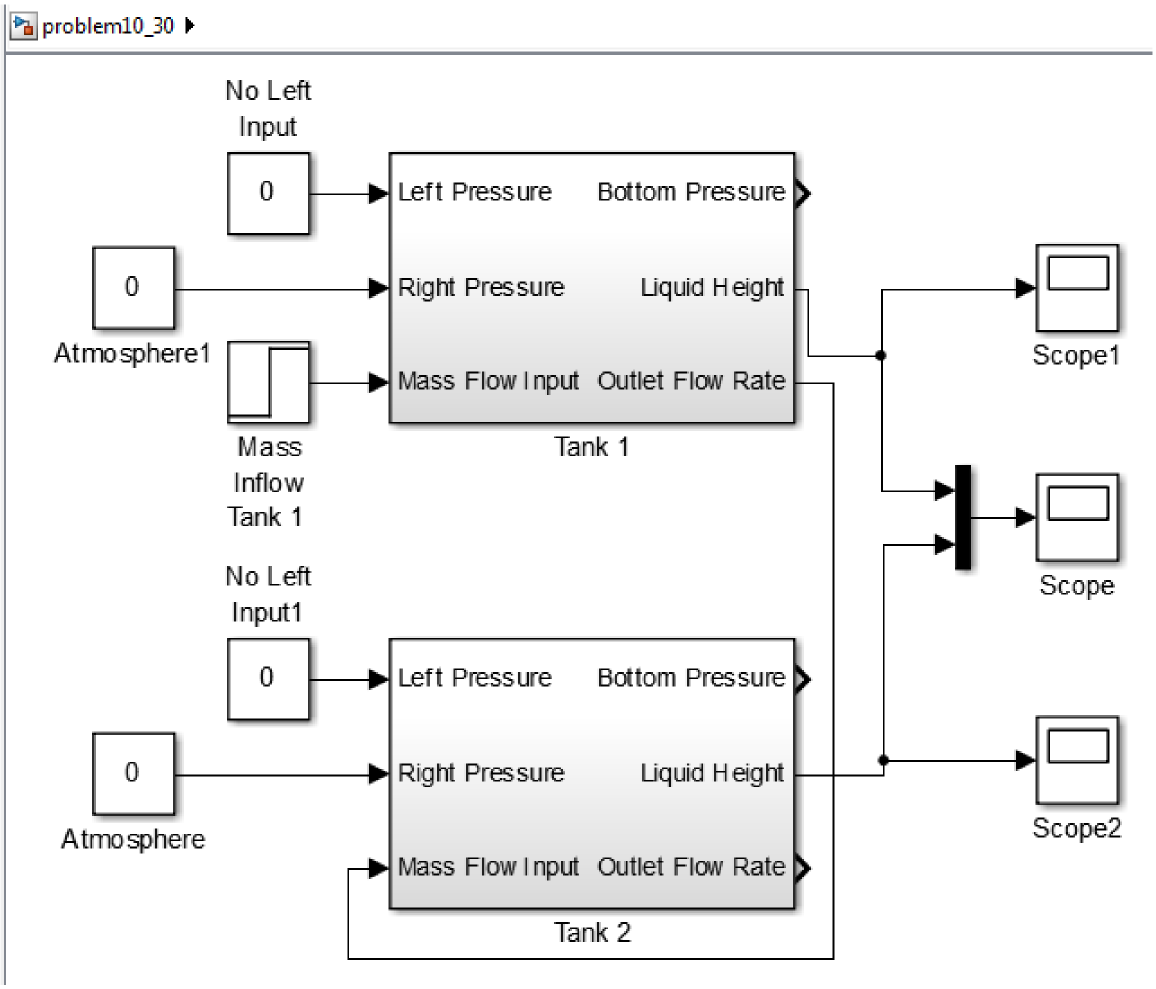
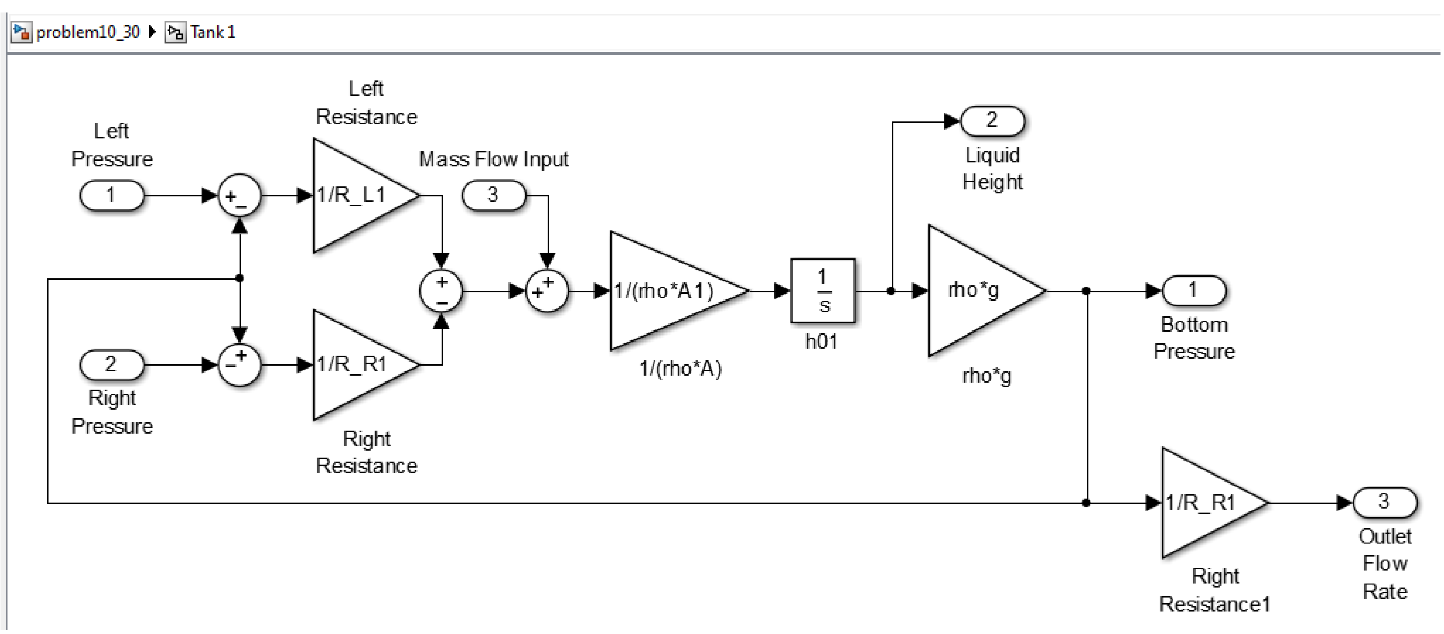
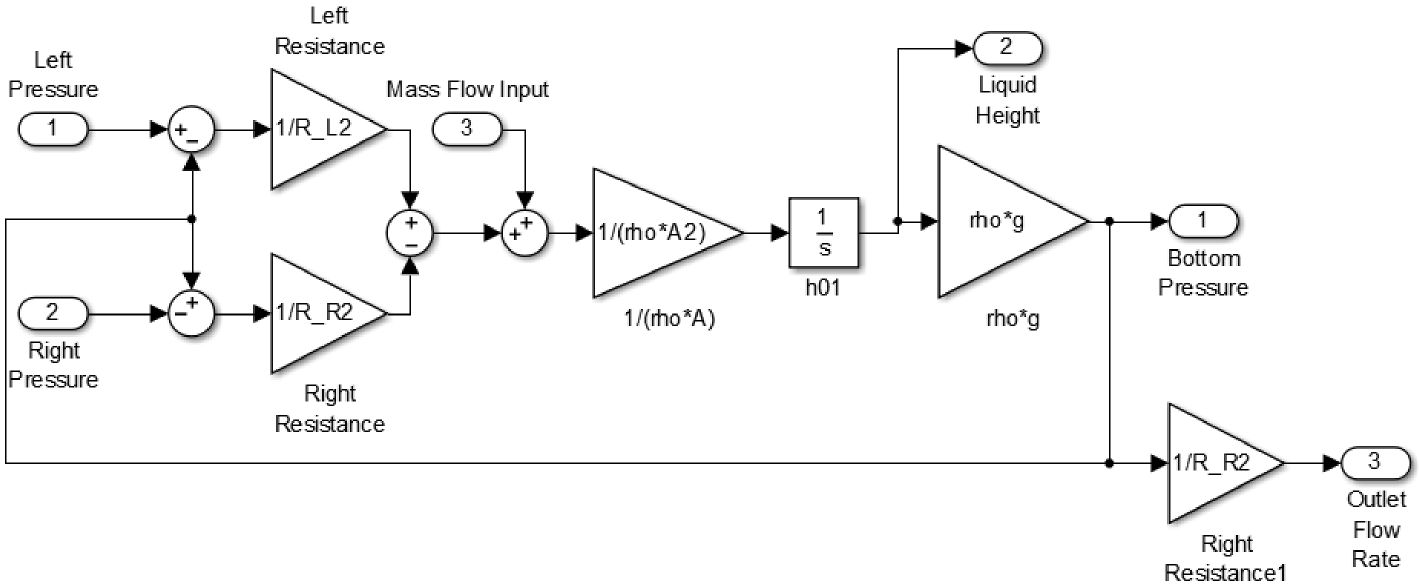


Figure P30


problem10_30
No Left Input
0
Left Pressure
Bottom Pressure
0
Atmosphere1
Right Pressure
Liquid Height
Scope1
Mass Flow Input
Outlet Flow Rate
Mass Inflow Tank 1
Tank 1
Scope
No Left Input1
0
Left Pressure
Bottom Pressure
0
Atmosphere
Right Pressure
Liquid Height
Scope2
Mass Flow Input
Outlet Flow Rate
Tank 2

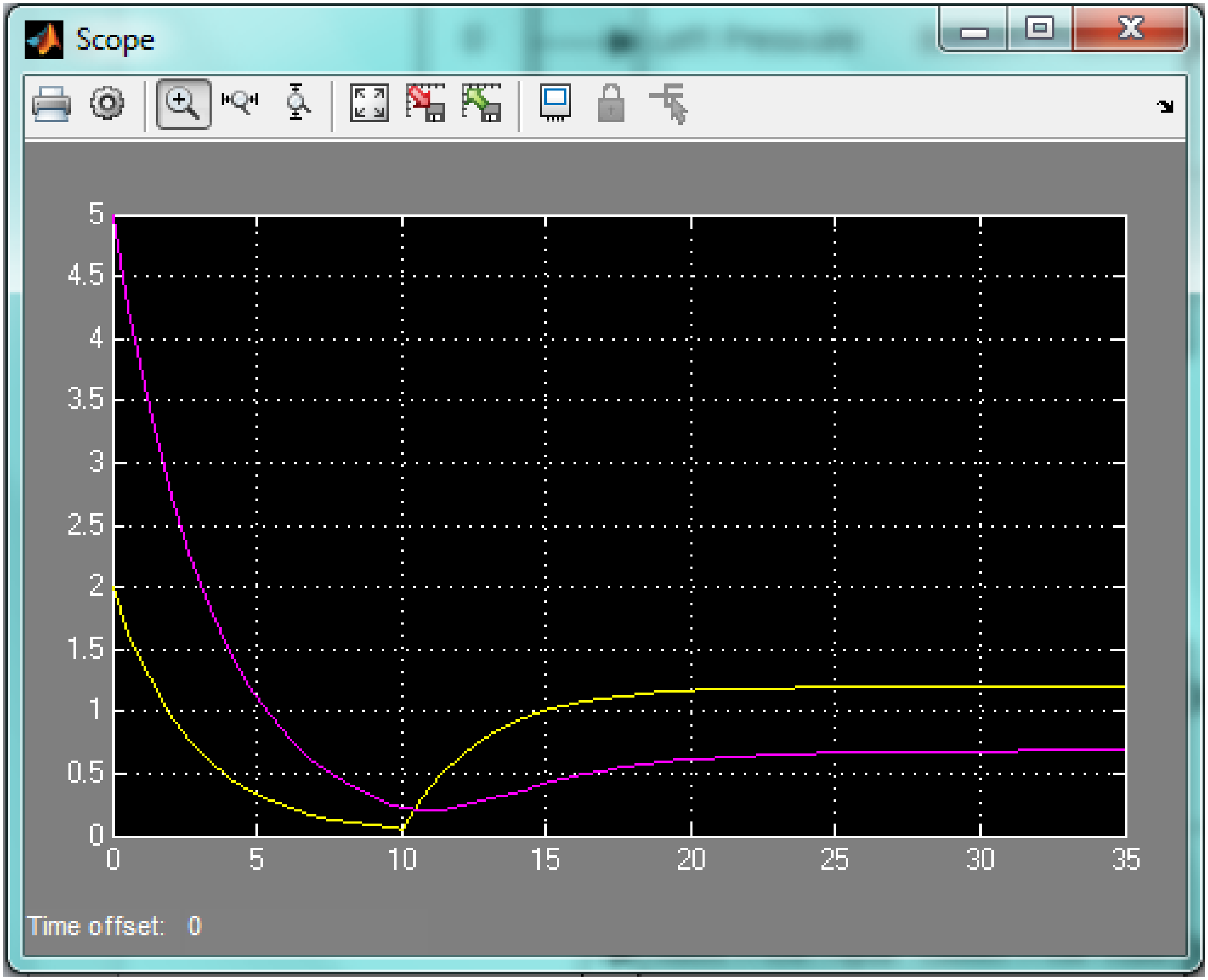

problem10_30
Tank 1
Left Resistance
Left Pressure
1
1/R_L1
Mass Flow Input
3
2
Liquid Height
1/(rho*A1)
1
s
rho*g
1
Bottom Pressure
h01
2
Right Pressure
1/R_R1
Right Resistance
1/(rho*A)
rho*g
1/R_R1
3
Right Resistance1
Outlet Flow Rate

problem10_30 ▸ Tank 2




```
% Problem 10.30 Script File

clc
clear

disp('Problem 10.30: Scott Thomas')

A1 = 3;% ft^2 Bottom Area of Tank 1
A2 = 5;% ft^2 Bottom Area of Tank 1
R_L1 = 1E6;% m^(-1)*s^(-1) No flow through the left-hand inlet
R_R1 = 30;% ft^(-1)*s^(-1)
R_L2 = 30;% ft^(-1)*s^(-1) One resistance between tanks: R_R1 = R_L2 = R_1
R_R2 = 40;% ft^(-1)*s^(-1)
rho = 1.94;% kg/m^3
g = 32.2;% m/s^2
h01 = 2;% ft Initial height in Tank 1
h02 = 5;% ft Initial height in Tank 1
qmi1 = 2.5;% kg/s Input flow into Tank 1
timestep1 = 10;% s Time when Input flow into Tank 1 is turned on
```

```
Problem 10.30: Scott Thomas
```


Scope
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0
5
10
15
20
25
30
35
Time offset: 0