PURVANG LAPSIWALA

UIN = 662689378

Submitted to Dr. Sabri Cetinkunt

University of Illinois at Chicago

Take Home Exam Report of ME511 Mechatronics II

Answer: 7.9

(a)

Given circuit diagram describes the control process for Hydraulic schematics for hand metering unit (HMU)steering system of wheel loader model WA450-5L by Komatsu. Here we have Emergency Steering System, which can be used when power (Load) demand is too high and normal system is not sufficient to provide enough power or main system fails.

Also there is one Joystick system which is used to give command to rotate steering and to rotate wheels and for that dedicated valve is allocated.

One pump is dedicated to the steering function; the other one is shared by the steering function and implement functions.

In a case of higher steering demand, flow from both pumps can be combined to serve the steering function. Main Valve is connected with the steering cylinder, which is moved based on to the side on which fluid moves so that desired result can be achieved.

(b)

The emergency backup steering system consist pump which is rotated by motor and fluid is coming from tank and it also consists of relief valve when pressure limit goes beyond the maximum pressure limit. This valve is connected with small valve which finally connected with main valve to control the movement of steering of system. This pump is activated whenever the pressure level in the steering line decreases to a preset value. This is necessary to guarantee steering control of the machine even in a case of failure of the main steering pump.

(c)

This machine can be steered either by a joystick or a steering wheel. The joystick actuates over the solenoids A and B which will modulate the flow to the pilots of the steering valve.

The steering wheel is based on an Orbitrol system. In Oritrol system oil from Pump flows into "P"-port of Orbitrol and then the quantity of oil in the proportion to the rotations of Steering Wheel flows into Steering Cylinders via "R"-port at right turn and via "L"-port at left turn. Accordingly, there are many advantages for designing of vehicles. The Orbitrol is used principally for hydrostatic power steering systems of industrial vehicles, construction vehicles and boats, but it can be used for some servo-type applications or any applications where visual positioning are required.

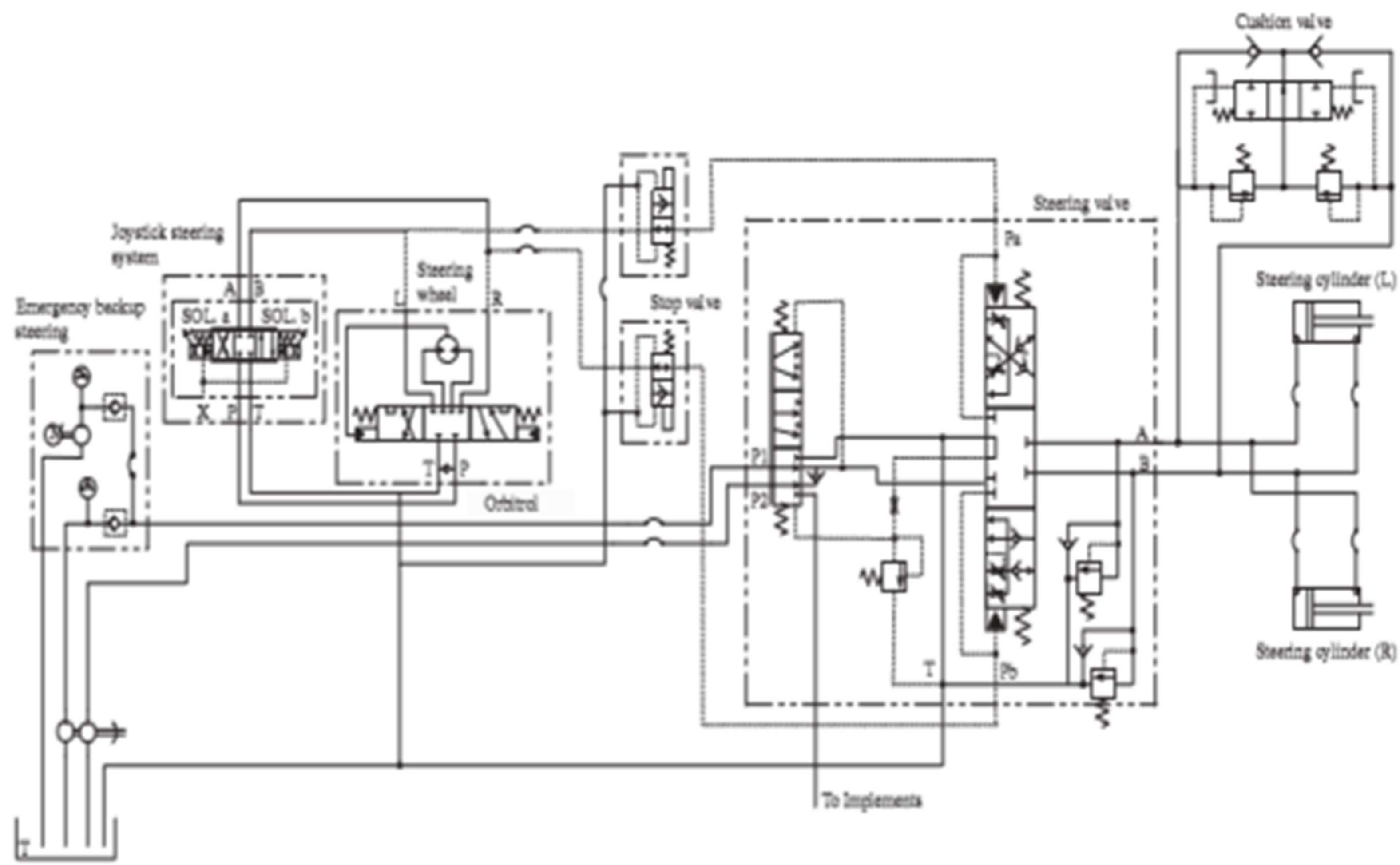
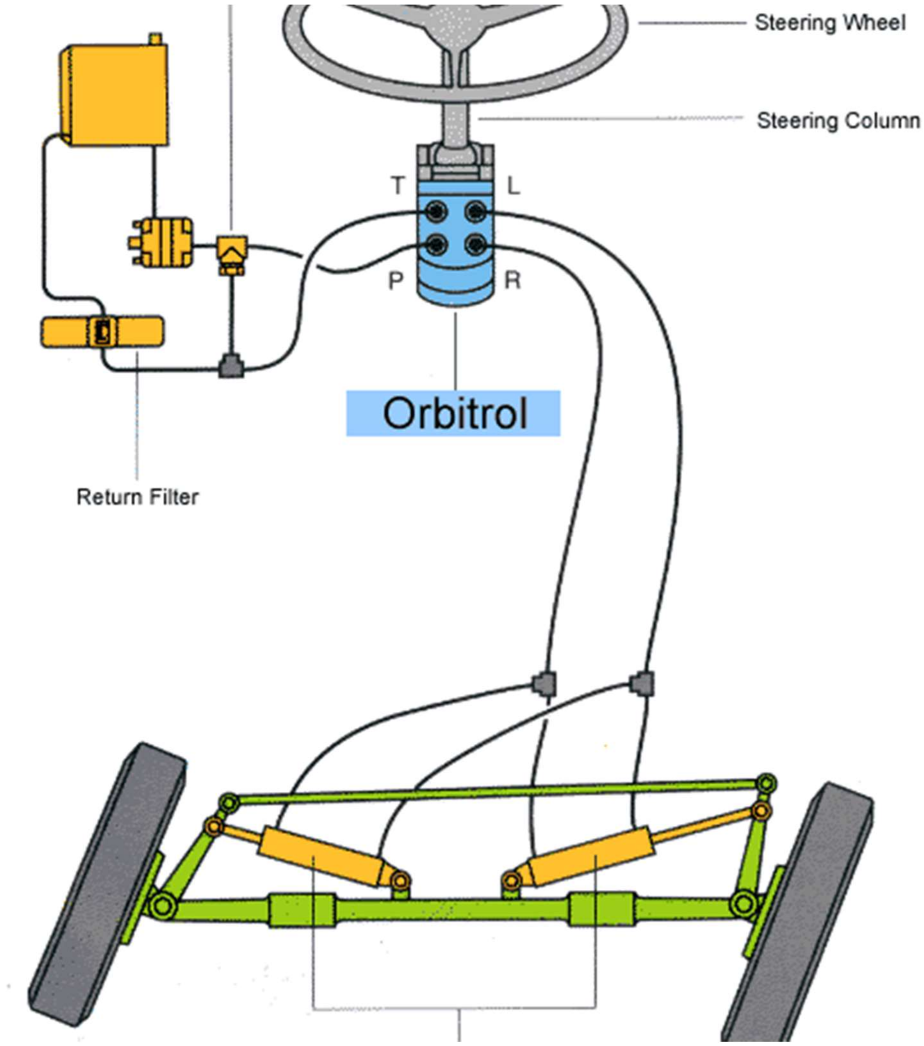


FIGURE 7.120: Hydraulic schematics for hand metering unit (HMU) steering system of wheel loader model WA450-5L by Komatsu.



Orbitol Steering System

Answer- 7.10

Some Contents are taken from mathworks Documents.

Reference of this problem is taken from textbook example matlab code for one axis Electro Hydraulic motion control system.

Equation for Dynamic Model of EH system is given. Also taken consideration of fluid compressibility and cylinder load inertia. When relief valve is active, fluid will act as ideal fluid and compressibility will not be taken in to account. Also given pump is close center pump hence Qpt=0. All the equations are given in example.

Also accumulator equations are given. Basic work of accumulator is to increase line capacitance and absorbs large pressure spikes and also it acts as power source for high demand or damper.

While designing relief valve, we also have to take see effects of area opening of valve. If it is too small then pressure will be high and Pp and Prelief will be same for most of time and vice versa. In first case user can achieve better resolution as we have very small opening and can be achieved high efficiency but problem will be there is huge pressure loss and also power loss as most of flow will go to relief valve.

On the other side, if we are keeping large valve area then we can't achieve better resolution and accuracy. So we have to take both consideration while designing.

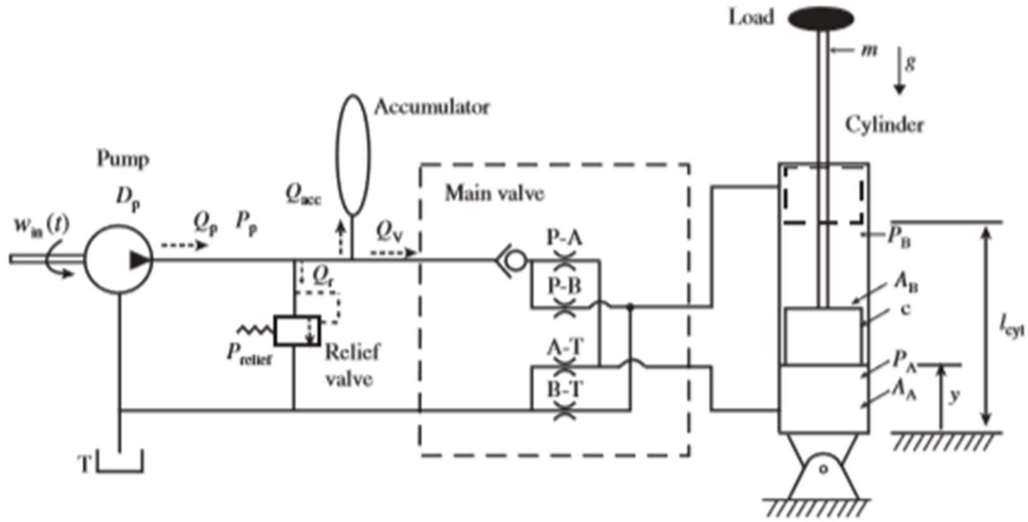


FIGURE 7.121: One degree of freedom hydraulic system with fixed displacement pump, closed center valve, a cylinder, and a relief valve.

<u>Case 1:</u>

Considerations while doing first part

Neglect cylinder and load inertia dynamics

Neglect fluid compressibility throughout hydraulic circuit

No accumulator

Ideal relief valve taking account of time delay and inertia effect

The only initial condition needed in this case is cylinder initial position. The rest of the initial conditions are not needed to be specified. They can be calculated directly from the simultaneous solution of the algebraic equations. In this case the differential equations describing the variation in cylinder pressure and cylinder motion reduces to an algebraic equation where the cylinder velocity is directly determined by the flow rate into the cylinder. The solution can be obtained directly by substitution among the algebraic equations since there are as many equations as there are unknowns.

However, a more useful method is to use a numerical method to solve these equations. For solution of algebraic equation , we need some initial values for other variables such as Pp, Pa,Pb,Y,Qr and from this values we can find roots of algorithms so initial guess/value is required to solve this problem. In order to simulate this case, simply set the left hand side of the differential equations (1)-(4) to zero, turning them into algebraic equations, and solve the whole equation set numerically.

So putting those value to main equation , equation behavior is non linear. Linearization that equation was done by matlab "fsolve" function. To linearize that function, I have take some reference from Internet for applying algorithm using optimset function to get value of x.

For simulation, set given equation to zero or any initial values and then we can easily solve any given differential equation numerically.

Matlab file for Case 1 without any consideration of dynamics and compressibility is below.

File for valve response as trapezoidal Command Signal, Flow rate and Relief Valve

```
%%% cyl_ae1.m

function F=cyl_ae1(x)
global D_p C_d x_db p_relief K_relief ;
global A_a A_b l_cyl mass1 c ;
global Q_p Q_pa Q_pb Q_at Q_bt Q_pt Q_r ;
global p_p p_a p_b p_t ;
global y ydot ;
global w_pump ;
global x_s F_load ;
ydot = x(1);  % velocity of cylinder
p_a = x(2);   % pressure on A side
p_b = x(3);   % pressure on B side
p_p = x(4);   % pump pressure

% Valve dynamics while considerinng Deadband and respecteed Area
if (abs(x_s) >= x_db )
    A_pa = ((20*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_pb = (( 10*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_at = ((40*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_bt = ((10*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
else
    A_pa = 0.0 ;
    A_pb = 0.0 ;
    A_at = 0.0 ;
    A_bt = 0.0 ;
end

% Flow rates for individual Section
Q_p = D_p * w_pump ;
Q_pt = 0.0 ; % closed center valve.
if(x_s >= 0.0 )
    Q_pa = C_d * A_pa * (abs(p_p - p_a))^0.5 ;
    Q_bt = C_d * A_bt * (abs(p_b - p_t))^0.5 ;
    Q_pb = 0.0 ;
    Q_at = 0.0 ;
else
    Q_pb = C_d * A_pb * (abs(p_p - p_b))^0.5 ;
    Q_at = C_d * A_at * (abs(p_b - p_t))^0.5 ;
    Q_pa = 0.0 ;
    Q_bt = 0.0 ;
end
Q_acc = 0.0 ;

% Ideal relief valve, without transient dynamics

if (p_p < p_relief)
    Q_r = 0.0 ;
    if x_s >= 0.0
        F = [ (- c * ydot + p_a * A_a - p_b *A_b - F_load) ;
            (Q_pa - ydot * A_a) *10^6 ; (-Q_bt + ydot * A_b)*10^6; (Q_p-(Q_pa + Q_pt +
Q_r + Q_acc))* 10^6] ;
    else
        F = [ (- c * ydot + p_a * A_a - p_b *A_b - F_load) ;
```

```
            (-Q_at - ydot * A_a)* 10^6 ; (Q_pb + ydot * A_b)* 10^6 ;
            (Q_p-(Q_pb + Q_pt + Q_r + Q_acc))* 10^6 ] ;
    end
else
    Q_r = K_relief * (p_p - p_relief) ;
    if x_s >= 0.0
        F = [(- c * ydot + p_a * A_a - p_b *A_b - F_load);
            (Q_pa - ydot * A_a)* 10^6 ; (-Q_bt + ydot * A_b)* 10^6 ;
            (Q_p-(Q_pa + Q_pt + Q_r + Q_acc))* 10^6 ] ;
    else
        F = [ (- c * ydot + p_a * A_a - p_b *A_b - F_load) ;
            (-Q_at - ydot * A_a)* 10^6 ; (Q_pb + ydot * A_b)* 10^6 ;
            (Q_p-(Q_pb + Q_pt + Q_r + Q_acc))* 10^6 ] ;
    end
end
return;
```

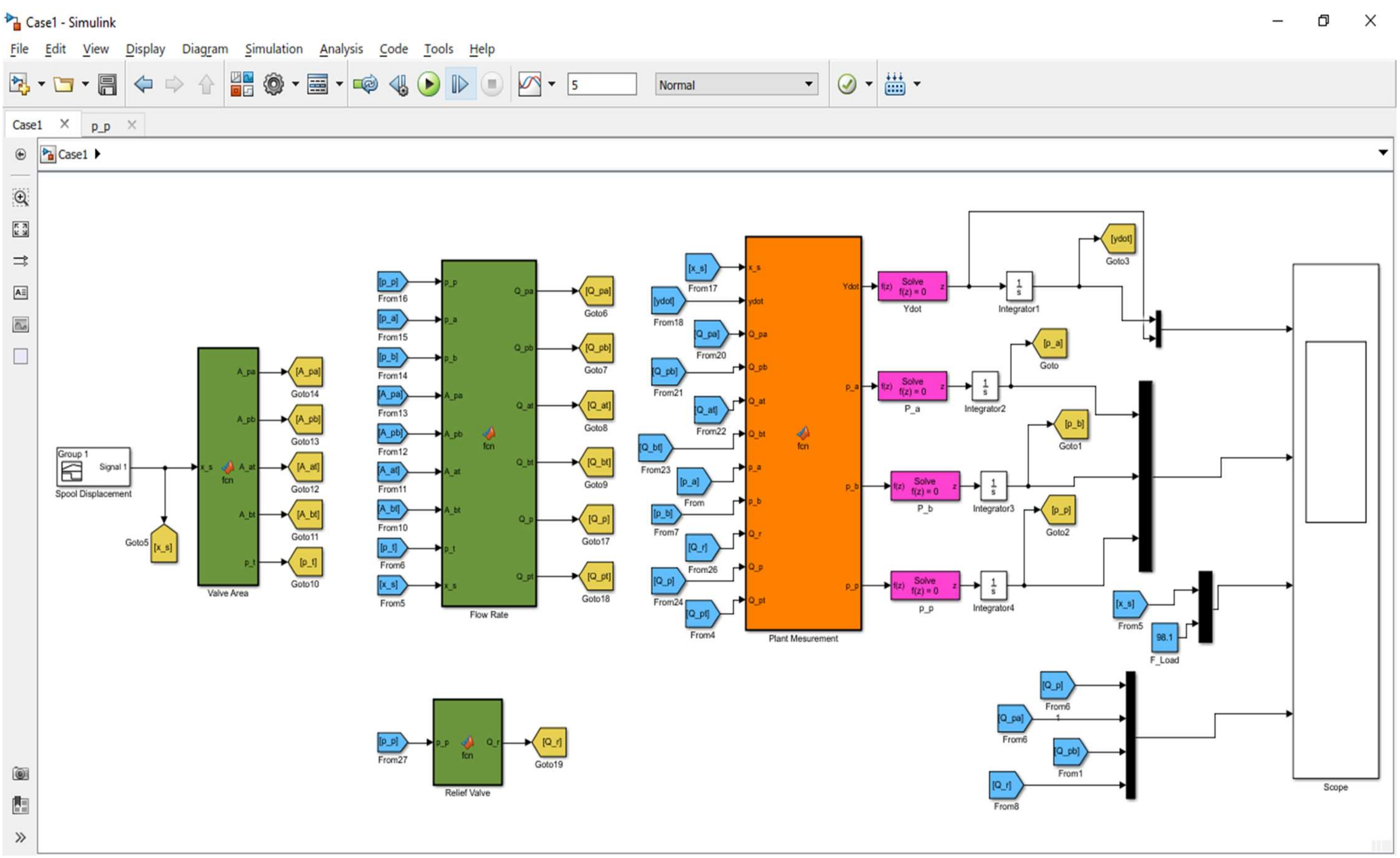
%%% File for output velocity of cylinder

```
% cyl_dyn1.m
function zdot=cyl_dyn1(t,z)
global y ydot ;
zdot(1) = ydot ;

%%% Main File

clc;
clear all;
Display("CASE 1")
global D_p C_d x_db p_relief K_relief ;
global A_a A_b l_cyl mass1 c ;
global Q_p Q_pa Q_pb Q_at Q_bt Q_pt Q_r ;
global p_p p_a p_b p_t ;
global y ydot ;
global w_pump ;
global x_s F_load ;

% Parameters
D_p = 0.0001 ; % Pump displacement volume per revolution
C_d = 0.065 ; % Valve parameters
x_db = 10 ; % Valve deadband in percentage of total travel

% Relief valve parameters
p_relief = 20 * 10^6 ; % [N/m^2] = Pa
K_relief = 0.01 * 10^-6 ;

% Cylinder parameters
A_a = 0.01 ;
A_b = 0.005 ;
l_cyl = 1.0 ; % [m]
mass1 = 10000 ; % [kg]
c = 0.0 ; % [N/(m/sec)]

% Input conditions
w_pump = 25 ; % Pump input shaft speed; [rev/sec]
F_load = mass1 * 9.81 ; % kg m/sec^2 = N

% Initial conditions on head and rod-end volume of cylinder.
y = 0.1 ; % Initial cylinder position
ydot = 0.0 ; % Initial cylinder velocity
p_p = p_relief ; % Pump pressure
```

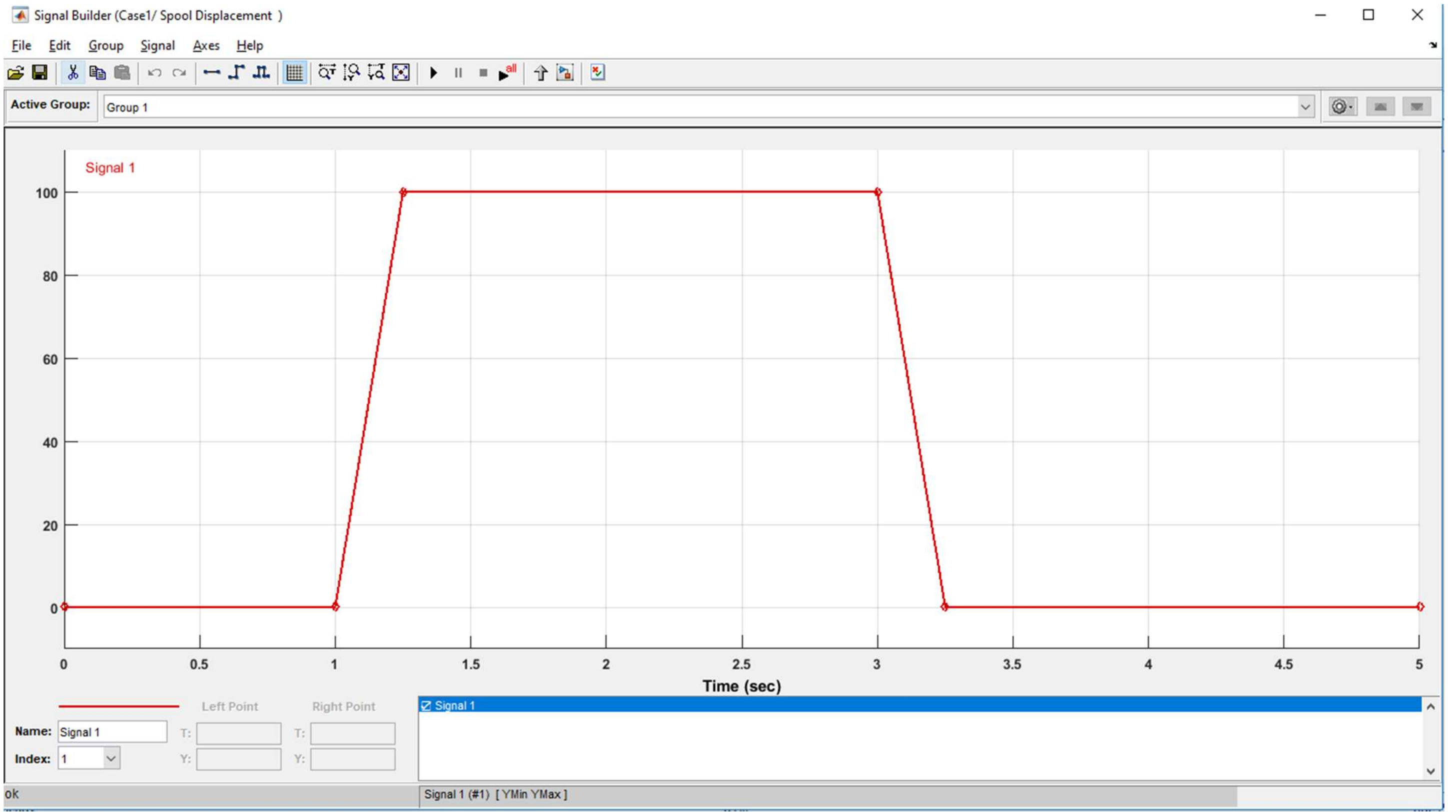
```
p_a = F_load/A_a ;
p_b = 0.0 ;
Q_r = D_p * w_pump ;
p_t = 0.0 ; % Tank pressure

t_0 =0.0;
t_f =5.0 ;
t_sample = 0.001 ;
z=zeros(1);
x =zeros(4) ;
z(1) = y ;
x(1) = ydot ;
x(2) = p_a ;
x(3) = p_b ;
x(4) = p_p;
z_out=[Q_p Q_pa Q_pb Q_at Q_bt Q_r x_s F_load/1000 y ydot p_a p_b p_p] ;

% Position of valve at different time as trapezoidal function
for (t=t_0: t_sample:t_f)
if  t<1.0
    x_s = 0.0 ;
elseif (t>= 1.0 && t<=1.25)
    x_s = (100/0.25) *(t-1.0) ;
elseif (t> 1.10 && t<=3.0)
    x_s = 100 ;
elseif (t> 3.0 && t<=3.25)
    x_s = 100 - (100/0.25) * (t - 3.0) ;
else
    x_s = 0.0 ;
end

% Solve Algebraic Equations....
% From Mathworks Documents
options = optimset('Algorithm','levenberg-marquardt','Display','off','TolFun',1e-
11,'TolX',1e-10); % Minimizing value of x using lm algorithem to get

% accurate solution (Global Minimum Concept)
x = fsolve('cyl_ae1',x,options) ; % Using above options for finding optimum value for x
ydot = x(1) ;   % velocity of cylinder
p_a = x(2) ;   % pressure on A side
p_b = x(3) ;   % pressure on B side
p_p = x(4) ;   % pump pressure

% Solve ODEs...
t_span=[t,t+t_sample] ;
[T,z1] = ode45('cyl_dyn1',t_span, z); % Solving for output
[m,n]=size(z1);
z(:)=[z1(m,:)] ;
y = z(1) ;
z_out=[ z_out;
        Q_p Q_pa Q_pb Q_at Q_bt Q_r x_s F_load/1000 y ydot p_a p_b p_p ] ;
end

[m,n]=size(z_out);
t_inc = (t_f-t_0)/m ;
tout=t_0:t_inc:t_f-t_inc;
tout = tout' ;

figure(1) ;
subplot(2,2,1) ;
plot(tout, z_out(:,1), 'k',tout, z_out(:,2), 'b',tout, z_out(:,5), 'm',tout, z_out(:,6),
'c');
```

```
xlabel('Time (sec)') ; ylabel('Flow rate (m^3/sec)') ;
legend('Q_p','Q_{pa}','Q_{bt}','Q_r');
subplot(2,2,2) ;
plot(tout, z_out(:,7), 'k',tout, z_out(:,8), 'b');
xlabel('Time (sec)') ; ylabel('Spool position and External Load') ;
legend('x_s','F_{load}');
subplot(2,2,3) ;
plot(tout, z_out(:,9) , 'k',tout, z_out(:,10), 'b');
xlabel('Time (sec)') ; ylabel('Cylinder position and velocity') ;
legend('y','ydot');
subplot(2,2,4) ;
plot(tout, z_out(:,11), 'k',tout, z_out(:,12), 'b',tout, z_out(:,13), 'g');
xlabel('Time (sec)') ; ylabel('Pressure [Pa]') ;
legend('p_a’,’p_b’,’p_p');
```

Simulink File For Case 1:

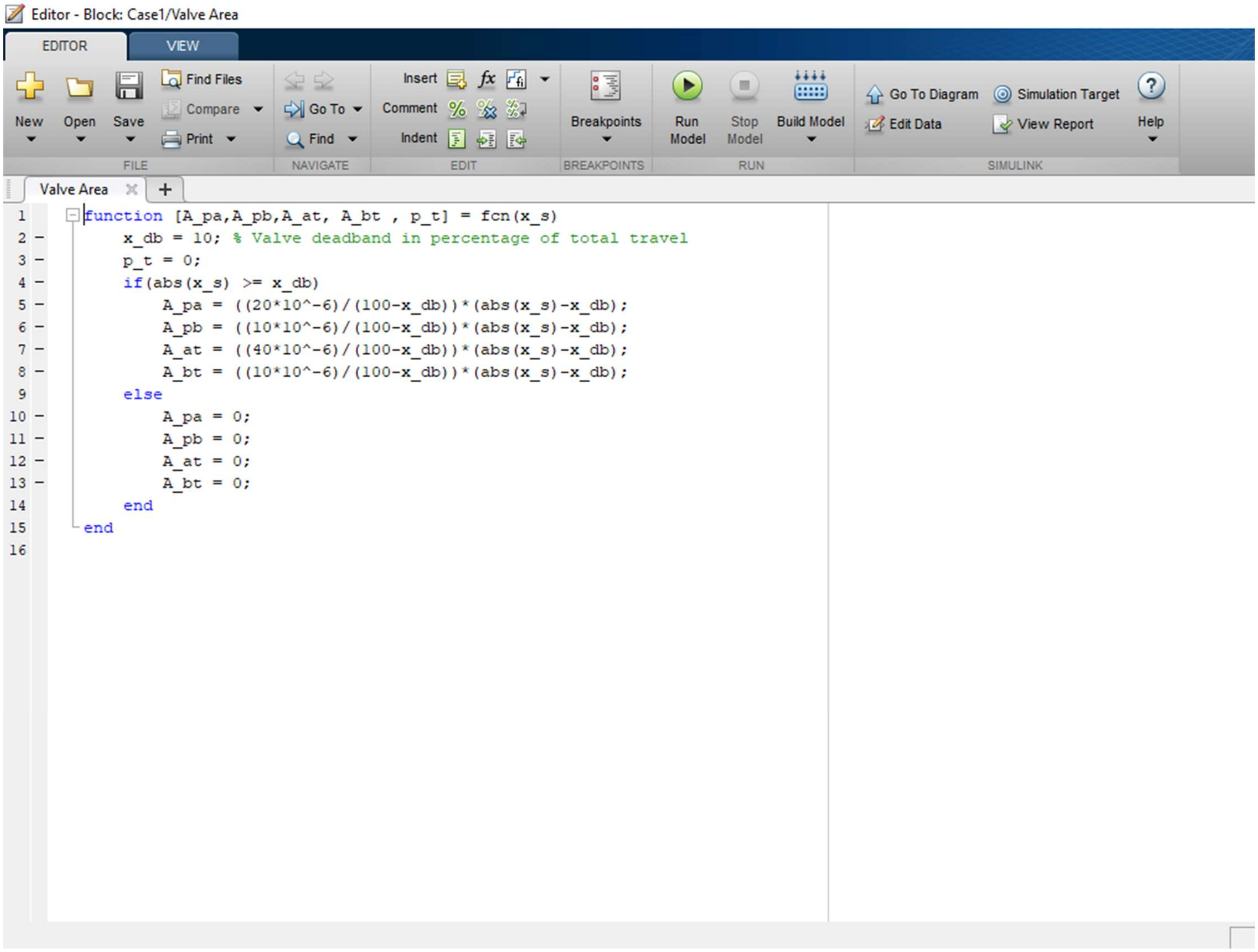


Case 1



Spool Displacement

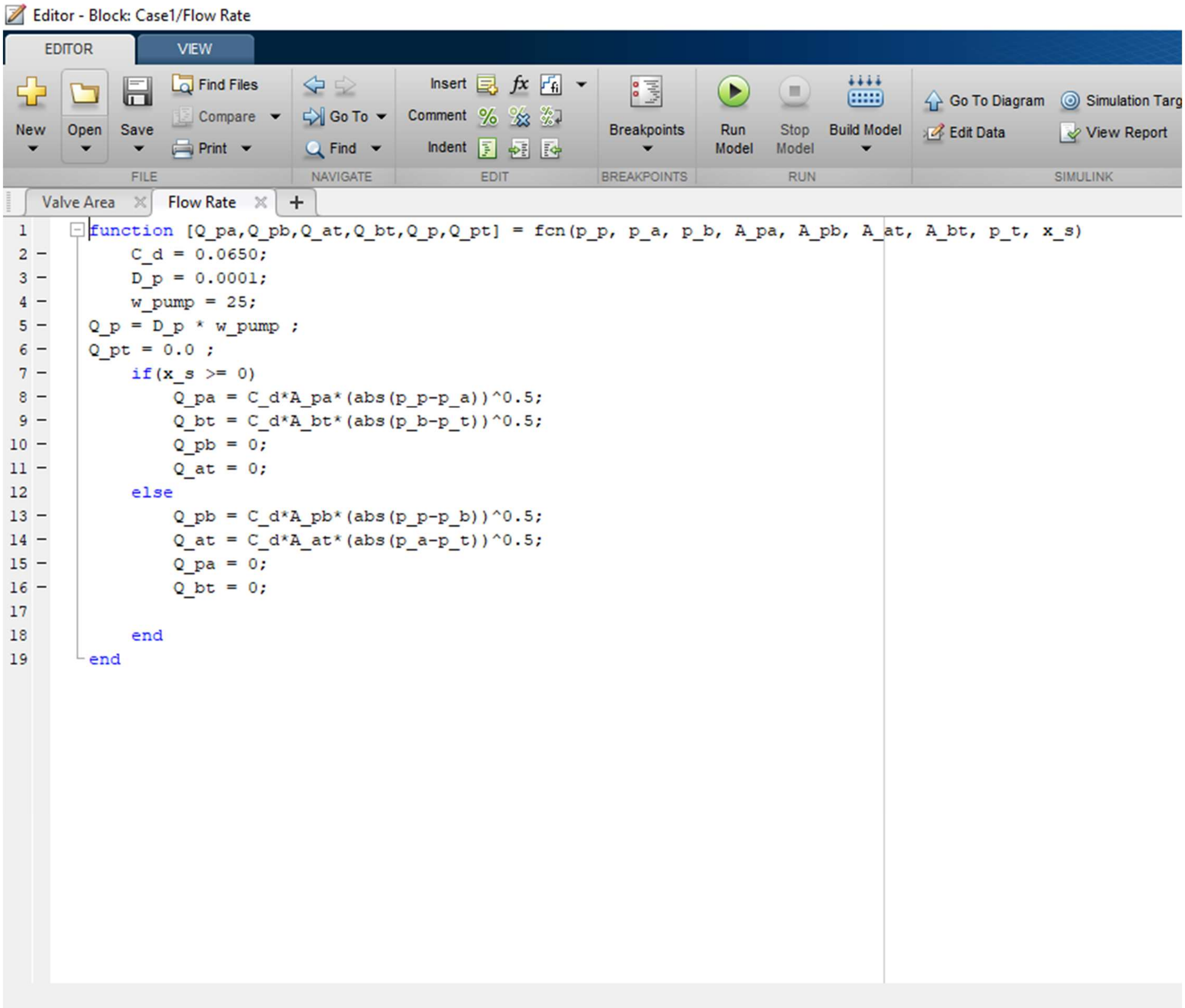
- Total simulation is carried out for 5 seconds.
- Valve spool displacement as given in question is taken as Trapezoidal signal. So Valve operates on Four different Modes.
- First between O to 1 second, spool is completely closed and there is no flow through Valve.
- After that, in 1 to 1.25 second, Flow is increasing proportionally and at 1.25 second it reaches to 100 percent
- After that, Flow remains constant until 3.25 second.
- Then flow starts decreasing from 3.0 to 3.25 second proportionally.
- And Flow again become zero after 3.25 to 5 second.




```
function [A_pa,A_pb,A_at, A_bt , p_t] = fcn(x_s)
    x_db = 10; % Valve deadband in percentage of total travel
    p_t = 0;
    if(abs(x_s) >= x_db)
        A_pa = ((20*10^-6)/(100-x_db))*(abs(x_s)-x_db);
        A_pb = ((10*10^-6)/(100-x_db))*(abs(x_s)-x_db);
        A_at = ((40*10^-6)/(100-x_db))*(abs(x_s)-x_db);
        A_bt = ((10*10^-6)/(100-x_db))*(abs(x_s)-x_db);
    else
        A_pa = 0;
        A_pb = 0;
        A_at = 0;
        A_bt = 0;
    end
end
```

Area function of spool displacement

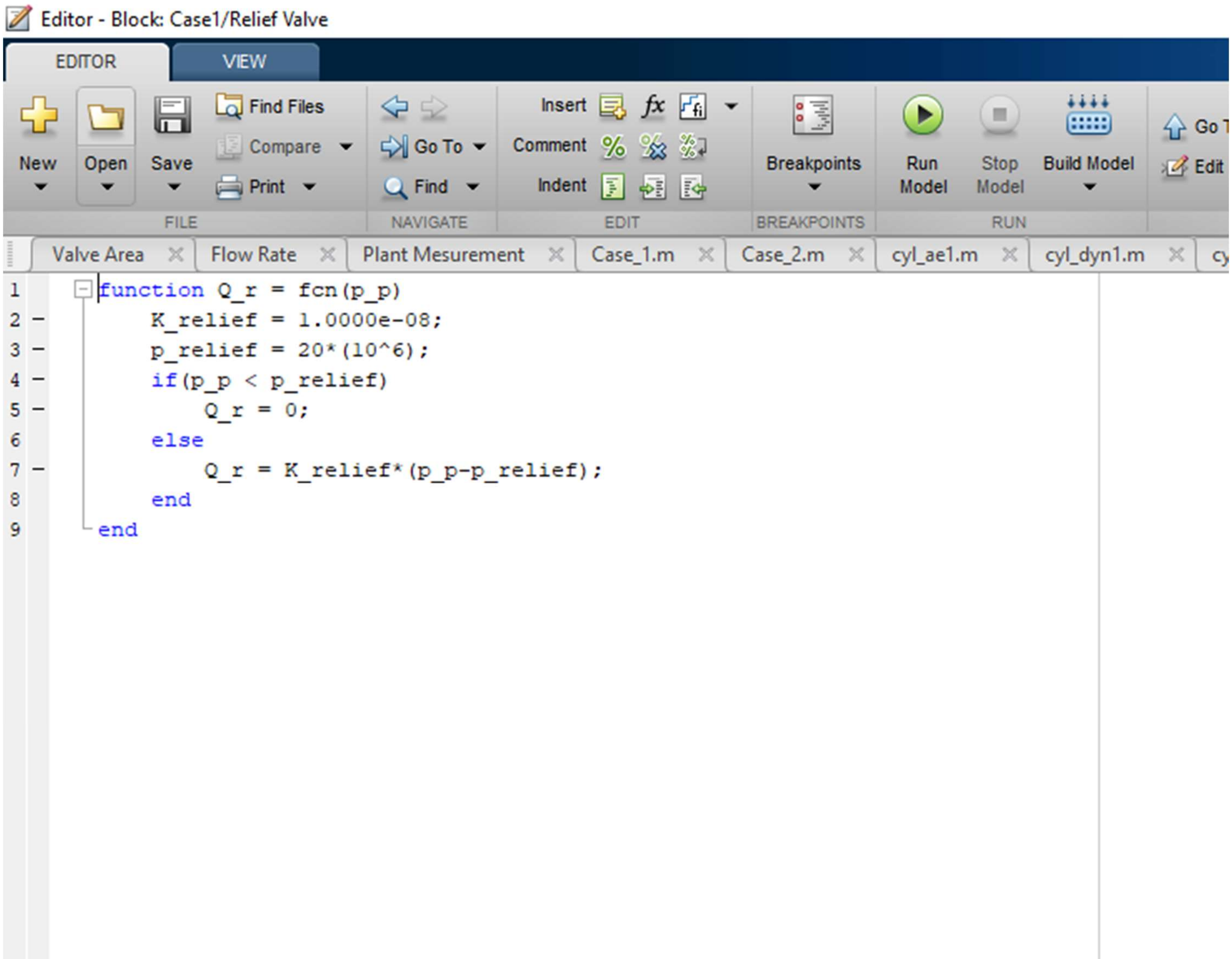
- This File shows Linear Relationship between Spool Displacement and Change in Area takes place as function of spool displacement.




```
function [Q_pa,Q_pb,Q_at,Q_bt,Q_p,Q_pt] = fcn(p_p, p_a, p_b, A_pa, A_pb, A_at, A_bt, p_t, x_s)
    C_d = 0.0650;
    D_p = 0.0001;
    w_pump = 25;
Q_p = D_p * w_pump ;
Q_pt = 0.0 ;
    if(x_s >= 0)
        Q_pa = C_d*A_pa*(abs(p_p-p_a))^0.5;
        Q_bt = C_d*A_bt*(abs(p_b-p_t))^0.5;
        Q_pb = 0;
        Q_at = 0;
    else
        Q_pb = C_d*A_pb*(abs(p_p-p_b))^0.5;
        Q_at = C_d*A_at*(abs(p_a-p_t))^0.5;
        Q_pa = 0;
        Q_bt = 0;

    end
end
```

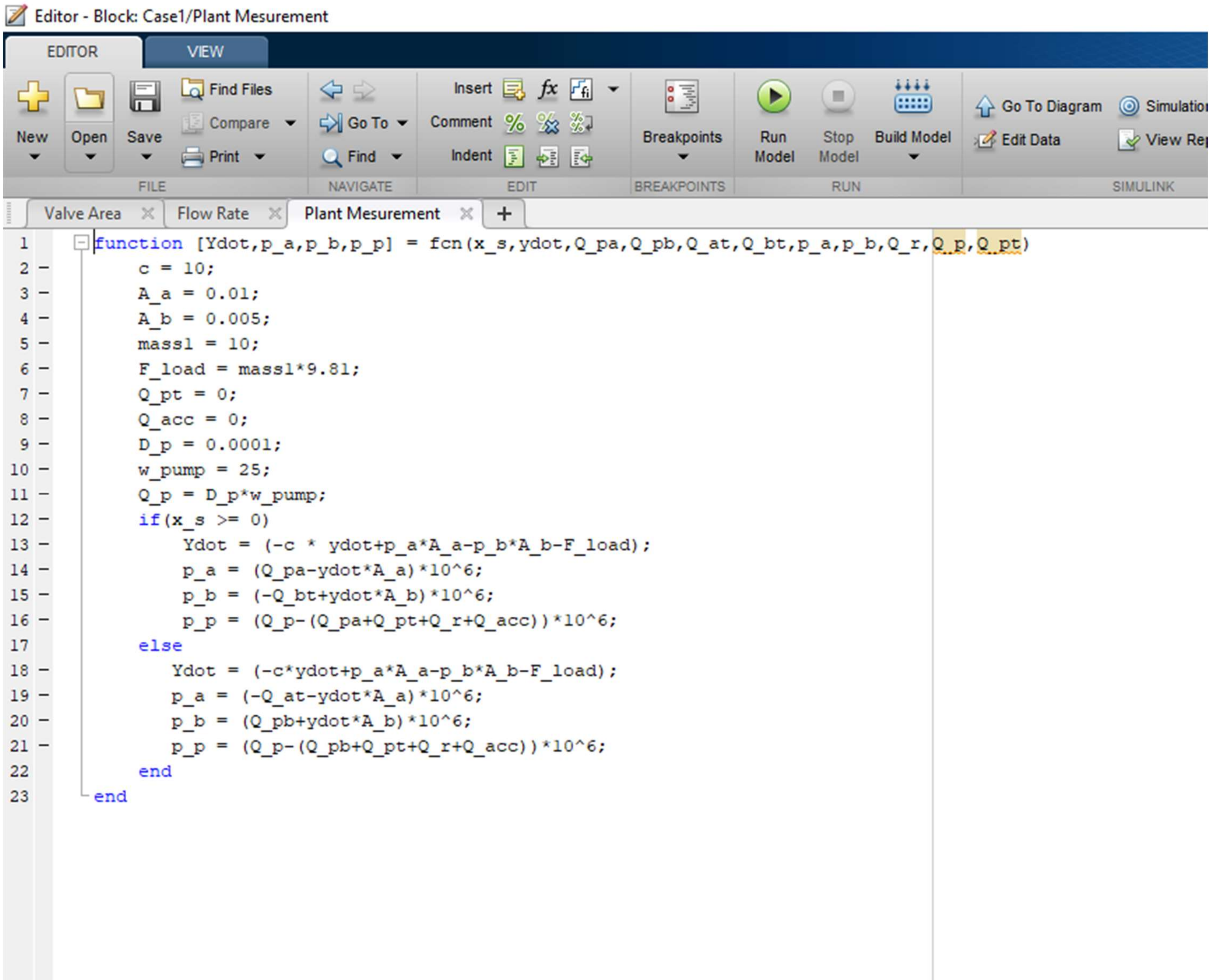
Flow Rates through Valve

- This file shows the flowrate to cylinder as function of Area that is open depends upon spool position.
- One Part is for Up motion of cylinder and other part is for down motion of cylinder.
- As spool position is greater than 0,there will be flow from pump to A side of cylinder so cylinder will go down side which will cause the flow from B side of cylinder to Tank and From pump to B side of Cylinder , there is zero flow rate because all flow goes to A side and same for A to tank.



Relief Valve Model

- This File shows how relief valve works and in which conditions , relief valve opens to dump excessive flow
- Basically when pump pressure us increases greater than relief valve pressure in our example it is given, Relief valve opens which basically puppet type valve and puppet moves its position because of excessive pressure and it dumps excessive flow to the Tank or else there is no flow through relief valve in normal condition.

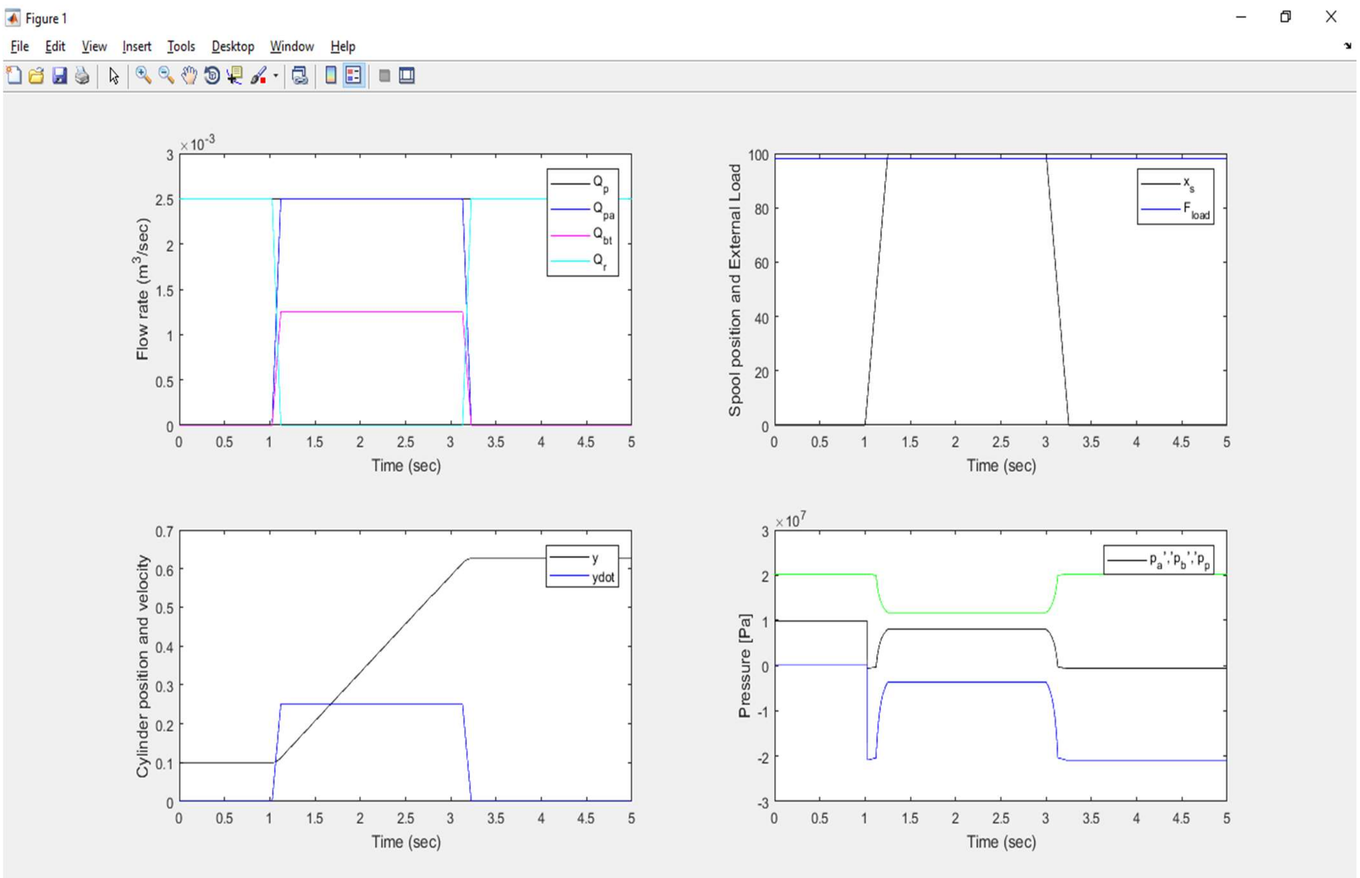


Plant Output Variable

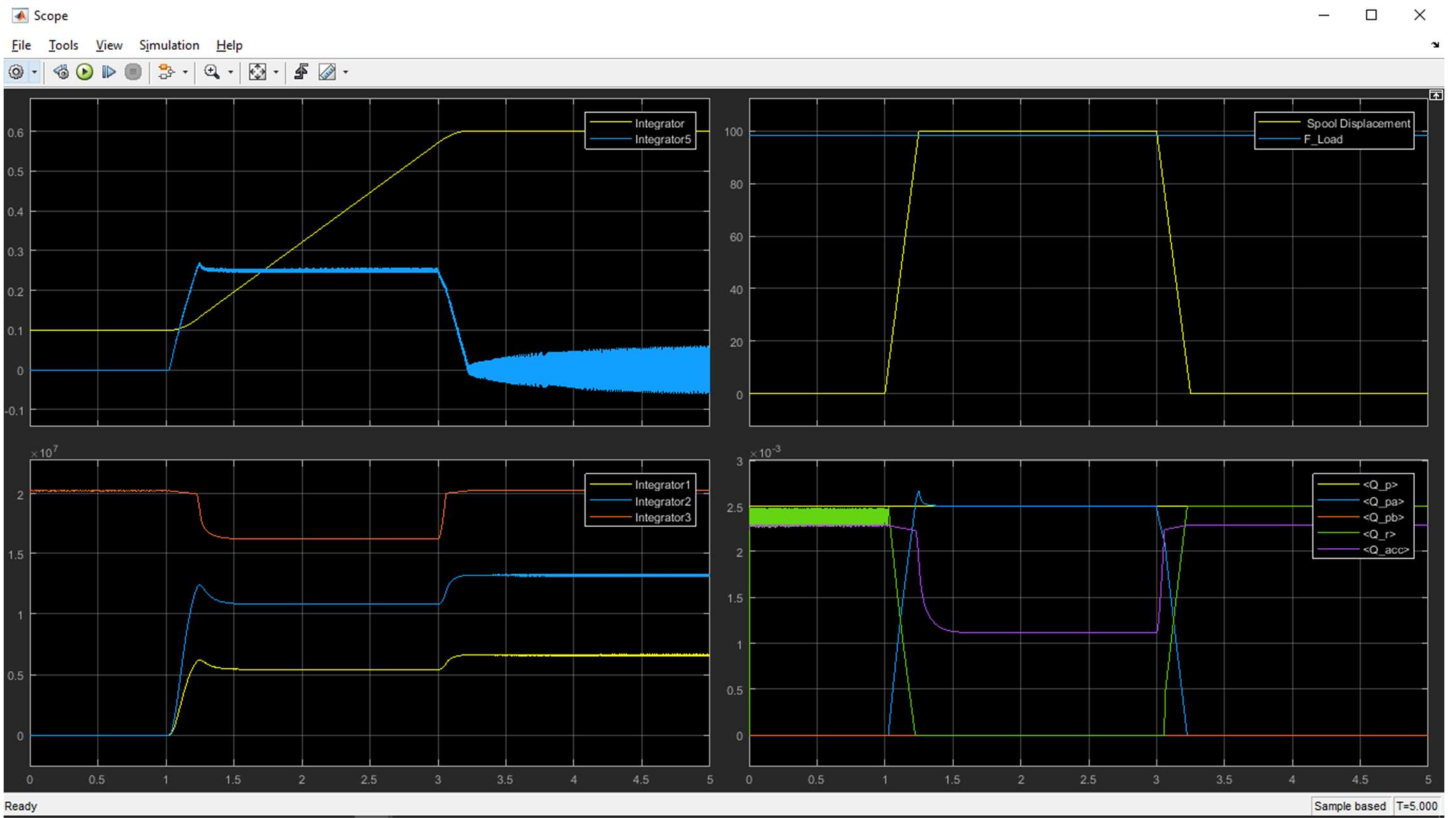
- This files shows how we can achieve desired output that we require.
- Interested parameters are Position and velocity of cylinder piston, Pressure at Side A, B and pump pressure.
- As it is closed loop control, so this system has some kind of position measurement sensors and pressure measurement sensors in order to feed back again for calculation.

Case 1 Results

Output From matlab file



Output from Simulink File



Observation

- Output from matlab and Simulink file almost are same. As given, there is no inclusion of Non linearity and dynamics as well as Accumulator so all changes in pressure becomes zero. So All Equations are reduced to be algebraic equation which can easily be solved numerically.
- Matlab is in built function called “fsolve” which uses particular algorithm called `levenberg-marquardt` algorithm and also uses tolerances, but in Simulink uses Trust Region based algorithm so this is the major reason why case result is bit different.

CASE 2:

Now In this case, Result is simulated after taking all things in to account such as cylinder and load inertial dynamics, fluid compressibility.

In this case we have the second order differential equation as equation, that defines the net force-acceleration relationship due to inertia and the differential equations. Now for given set of differential equations to be solved, we need initial condition of parameters which we neglected in case 1. All of the algebraic equations can be solved directly by using the available initial conditions on pressures.

We simulate this condition without the accumulator and with the accumulator in the circuit. Accumulator is added to the circuit model by including the Vacc and Qacc terms and the equations describing the dynamics of the accumulator state, pacc(t) and Vacc. Also we have to specify accumulator parameters such as Pmax, Pmin, Ppre and Vdisch.

Matlab file for Case 2 with consideration of dynamics and compressibility is below.

%% Consideration of Dynamics and Accumulator

```
function zdot=cyl_dyn2(t,z)
global D_p C_d x_db p_relief tau_relief K_relief ;
global p_max p_min p_pre V_disch K_acc C_acc;
global A_a A_b l_cyl mass1 c ;
global beta V_hose_pv V_hose_va V_hose_vb ;
global Q_p Q_pa Q_pb Q_at Q_bt Q_pt Q_r Q_acc ;
global p_p p_a p_b p_t p_acc V_acc ;
global y ydot ;
global w_pump ;
global x_s F_load ;
y = z(1)  ;  % Cylinder Position
ydot = z(2) ; % Velocity
p_a = z(3) ;  % A side pressure
p_b = z(4) ;  % B side pressure
p_p = z(5) ;  % pump pressure
p_acc = z(6) ; % Accumulator Pressure
V_acc = z(7) ; %Accumulator Volume


if (abs(x_s) >= x_db )
    A_pa = ((20*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_pb = (( 10*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_at = ((40*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
    A_bt = ((10*10^-6)/(100-x_db))* (abs(x_s) - x_db) ;
else
    A_pa = 0.0 ;
    A_pb = 0.0 ;
    A_at = 0.0 ;
    A_bt = 0.0 ;
end

% Flow rates
Q_p = D_p * w_pump ;
Q_pt = 0.0 ; % closed center valve.
if(x_s >= 0.0 )
    Q_pa = C_d * A_pa * (abs(p_p - p_a))^0.5 ;
    Q_bt = C_d * A_bt * (abs(p_b - p_t))^0.5 ;
    Q_pb = 0.0 ;
    Q_at = 0.0 ;
else
```

```
    Q_pb = C_d * A_pb * (abs(p_p - p_b))^0.5 ;
    Q_at = C_d * A_at * (abs(p_b - p_t))^0.5 ;
    Q_pa = 0.0 ;
    Q_bt = 0.0 ;
end

% Case 2: with Accumulator
if  V_acc <= 0.0 && p_p < p_acc
    Q_acc = 0.0;
else
    Q_acc = sign(p_p - p_acc) * K_acc * (abs(p_p - p_acc))^0.5 ;
end

% Non-Ideal relief valve, no transient dynamics
if  p_p < p_relief
    Q_r = 0.0 ;
else
    Q_r = K_relief * (p_p - p_relief) ;
end
zdot=zeros(7,1) ;

% ODEs...
zdot(1) = ydot ;

zdot(2) = (1/mass1)*(-c * ydot + p_a * A_a - p_b * A_b - F_load) ;
if x_s >= 0.0
    zdot(3) = (beta/(V_hose_va + y * A_a))*(Q_pa - ydot * A_a) ;
    zdot(4) = (beta/(V_hose_vb + (l_cyl - y) * A_b))*(-Q_bt + ydot * A_b);
    zdot(5) = (beta/(V_hose_pv + V_acc))*(Q_p-(Q_pa + Q_pt + Q_r + Q_acc));
else
    zdot(3) = (beta/(V_hose_va + y * A_a))*(-Q_at - ydot * A_a) ;
    zdot(4) = (beta/(V_hose_vb + (l_cyl - y) * A_b))*(Q_pb + ydot * A_b);
    zdot(5) = (beta/(V_hose_pv + V_acc))*(Q_p-(Q_pb + Q_pt + Q_r + Q_acc));
end
zdot(6) = (1/C_acc)* Q_acc ;
zdot(7) = Q_acc;
return;
```

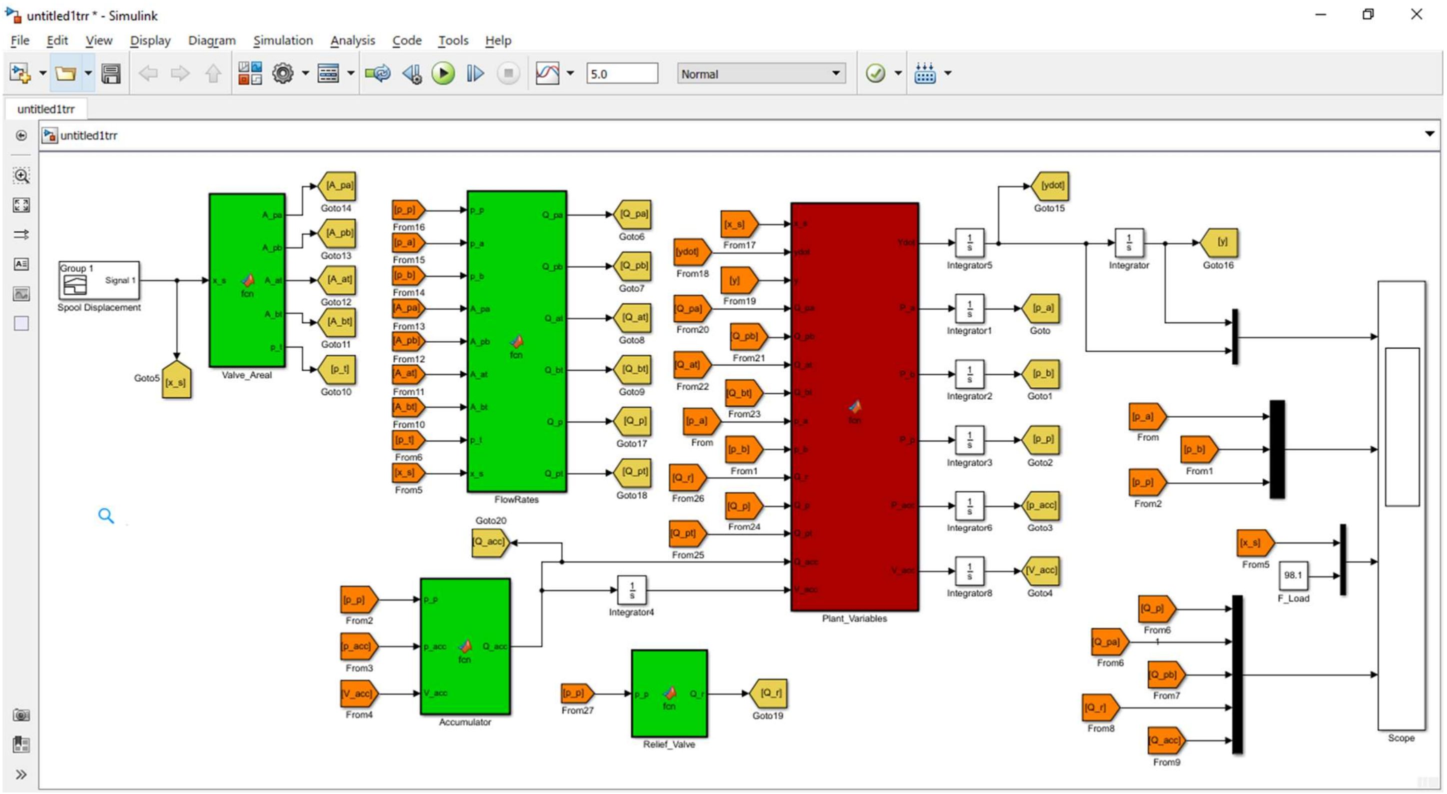
%% Main File

```
display("Case-2")
 % For CASE 2 %
global D_p C_d x_db p_relief tau_relief K_relief ;
global p_max p_min p_pre V_disch K_acc C_acc;
global A_a A_b l_cyl mass1 c ;
global beta V_hose_pv V_hose_va V_hose_vb ;
global Q_p Q_pa Q_pb Q_at Q_bt Q_pt Q_r Q_acc ;
global p_p p_a p_b p_t p_acc V_acc ; global y ydot ;
global w_pump ;
global x_s F_load ;
% Parameters
D_p = 0.0001 ; % Pump displacement volume per revolution

C_d = 0.065 ; % Valve parameters
x_db = 10 ; % Valve deadband in percentage of total travel
% Relief valve parameters
p_relief = 20 * 10^6 ; % [N/m^2] = Pa
tau_relief = 0.025 ; % time constant of relief valve dynamics
K_relief = 0.01*10^-6 ;
% Accumulator parameters
p_max = 20*10^6 ; % [N/m^2] = Pa
```

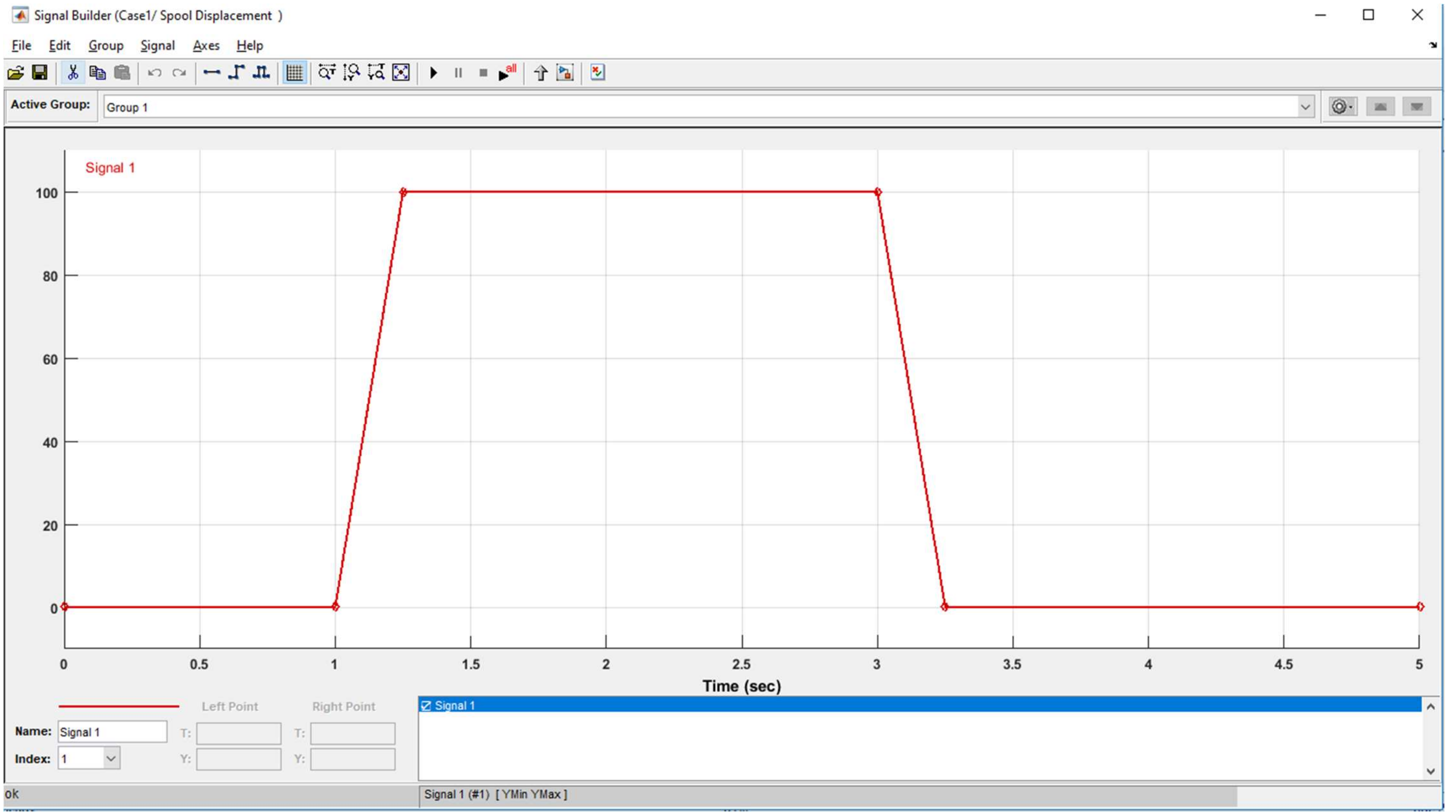
```
p_min = 15*10^6 ; % [N/m^2] = Pa
p_pre = 15*10^6 ; % [N/m^2] = Pa
V_disch = 0.005 ; % [m^3]
V_acc = 0.0 ; % initial fluid volume in the accumulator
K_acc = 1.0*10^-6 ;
C_acc = V_disch/(p_max - p_min) ;

% Cylinder parameters
A_a = 0.01 ;
A_b = 0.005 ;
l_cyl = 1.0 ; % [m]
mass1 = 10000 ; % [kg]
c = 10.0 ; % [N/(m/sec)]
% Fluid parameters and hose volumes
beta = 15.0*(10^8) ; % Bulk modulus
V_hose_pv = 0.0001 ; % [m^3]
V_hose_va = 0.0001 ; % [m^3]
V_hose_vb = 0.0001 ; % [m^3]
% Input conditions
w_pump = 25 ; % Pump input shaft speed; [rev/sec]
F_load = mass1 * 9.81 ; % kg m/sec^2 = N
% Initial conditions on head and rod-end volume of cylinder.
y = 0.1 ; % Initial cylinder position
ydot = 0.0 ; % Initial cylinder velocity
p_p = 0.0 ; % Pump pressure
p_a = F_load/A_a ;
p_b = 0.0 ;
p_acc = p_min ;
V_acc = 0.0 ;
Q_r = 0.0 ;
p_t = 0.0 ; % Tank pressure

t_0 =0.0;
t_f =5.0 ;
t_sample = 0.001 ;
z=zeros(7,1);
z(1) = y ; % Position of cylinder
z(2) = ydot ; % Velocity of cylinder
z(3) = p_a ; % Pressure at side A
z(4) = p_b ; % Pressure at side B
z(5) = p_p;  % Pump pressure
z(6) = p_acc ; % Accumulator Pressure
z(7) = V_acc ; % Accumulator Volume

z_out=[Q_p Q_pa Q_pb Q_at Q_bt Q_r Q_acc x_s F_load/1000 z(1) z(2) z(3) z(4) z(5)
z(6)/10^7 z(7)*1000 ] ; % Output Variables
for (t=t_0: t_sample:t_f)
    % Valve Position as function of trapezoidal signal
if t<1.0
    x_s = 0.0 ;
elseif (t>= 1.0 && t<=1.25)
    x_s = (100/0.25) *(t-1.0) ;
elseif (t> 1.10 && t<=3.0)
    x_s = 100 ;
elseif (t> 3.0 && t<=3.25)
    x_s = 100 - (100/0.25) * (t - 3.0) ;
else
    x_s = 0.0 ;
end

% Solve ODEs...
t_span=[t,t+t_sample] ;
```

```
[T,z1] = ode45('cyl_dyn2',t_span, z);
[m,n]=size(z1);
z(:)=[z1(m,:)] ;
y = z(1) ;
ydot = z(2) ;
p_a = z(3) ;
p_b = z(4) ;
p_p = z(5) ;
p_acc = z(6) ;
V_acc = z(7) ;
z_out=[z_out;
       Q_p Q_pa Q_pb Q_at Q_bt Q_r Q_acc x_s F_load/1000 z(1) z(2) z(3) z(4) z(5)
z(6)/10^7 z(7)*1000 ] ;
end
[m,n]=size(z_out);
t_inc = (t_f-t_0)/m ;
tout=t_0:t_inc:t_f-t_inc;
tout = tout' ;

figure(2) ;
subplot(3,2,1) ;
plot(tout, z_out(:,1), 'k',tout, z_out(:,2), 'b',tout, z_out(:,5), 'm', tout, z_out(:,6),
'c',tout, z_out(:,7), 'g');
xlabel('Time (sec)') ; ylabel('Flow rate (m^3/sec)') ;
legend('Q_p','Q_{pa}','Q_{bt}','Q_r', 'Q_{acc}');
subplot(3,2,2) ;
plot(tout, z_out(:,8), 'k',tout, z_out(:,9), 'b');
xlabel('Time (sec)') ; ylabel('Spool position and External Load') ;
legend('x_s','F_{load}');
subplot(3,2,3) ;
plot(tout, z_out(:,10) , 'k',tout, z_out(:,11), 'b');
xlabel('Time (sec)') ; ylabel('Cylinder position and velocity') ;
legend('y','ydot');
subplot(3,2,4) ;
plot(tout, z_out(:,12), 'k',tout, z_out(:,13), 'b',tout, z_out(:,14), 'g');
xlabel('Time (sec)') ;
ylabel('Pressure [Pa]') ;
legend('p_a','p_b','p_s');
subplot(3,2,5) ;
plot(tout, z_out(:,15), 'k',tout, z_out(:,16), 'b');
xlabel('Time (sec)') ;ylabel('Accumulator state') ;
legend('Pressure/10^7','Fluid Volume (liter)');
```

Simulink File

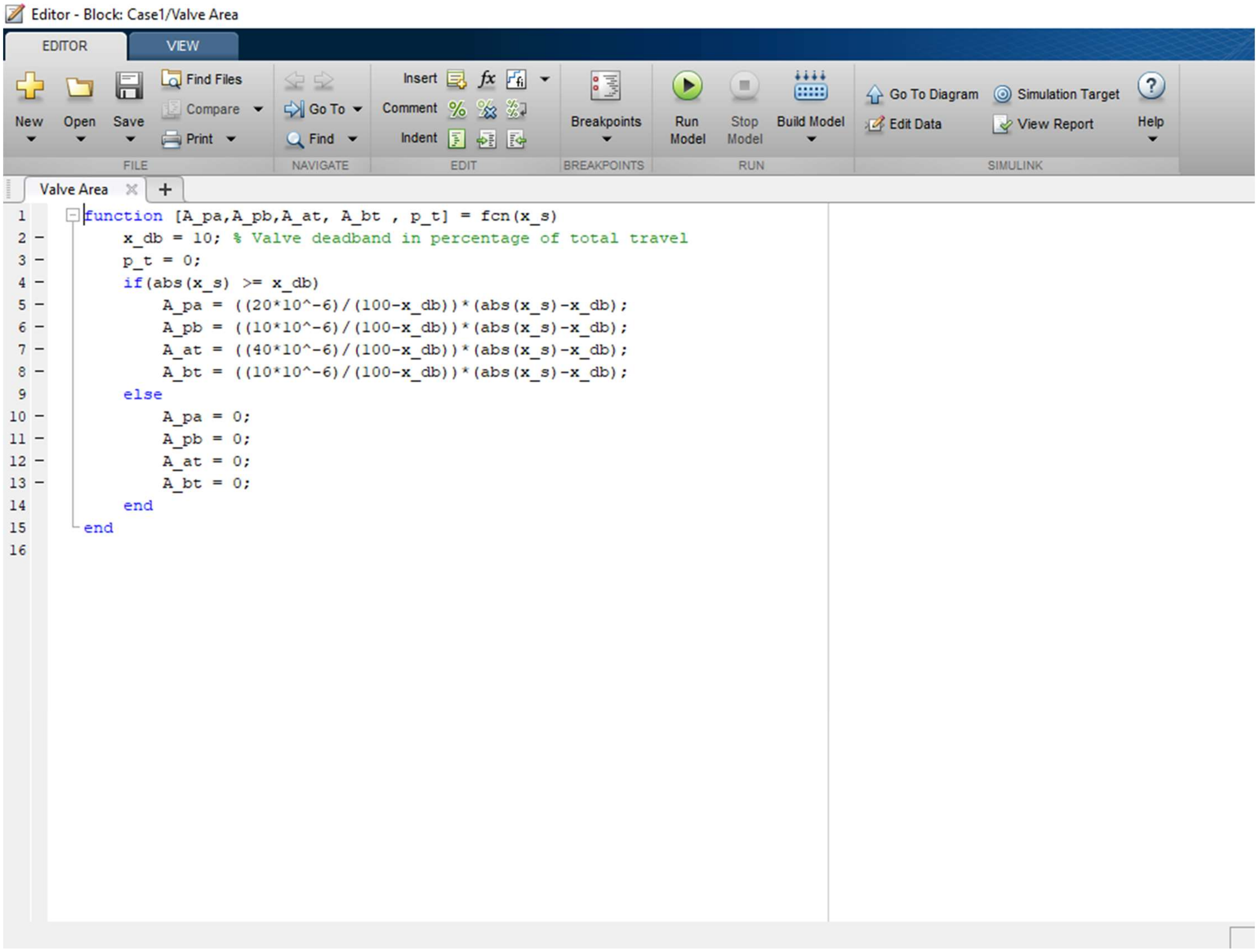


Simulink FIle



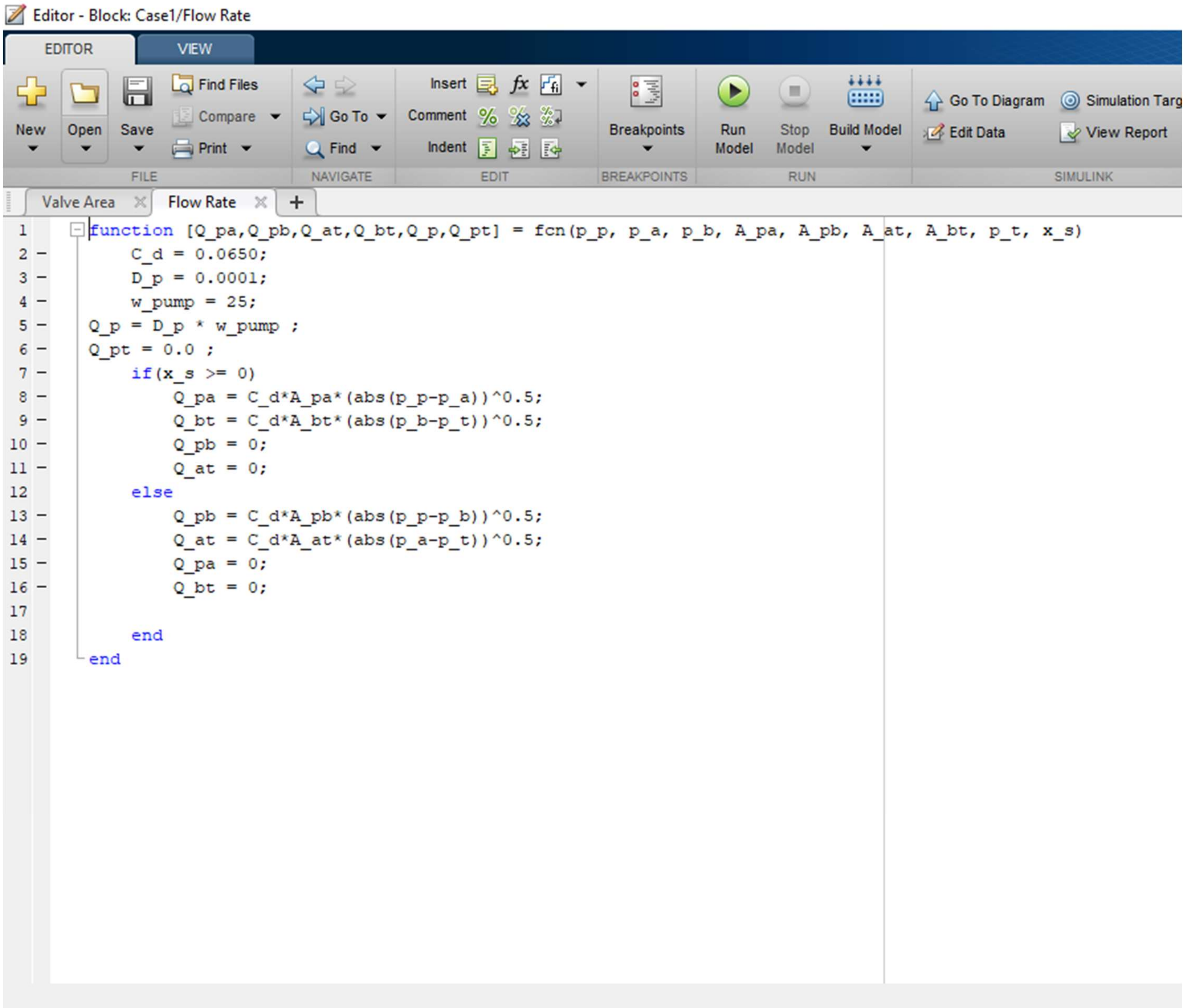
Spool Displacement

- Total simulation is carried out for 5 seconds.
- Valve spool displacement as given in question is taken as Trapezoidal signal. So Valve operates on Four different Modes.
- First between O to 1 second, spool is completely closed and there is no flow through Valve.
- After that, in 1 to 1.25 second, Flow is increasing proportionally and at 1.25 second it reaches to 100 percent
- After that, Flow remains constant until 3.25 second.
- Then flow starts decreasing from 3.0 to 3.25 second proportionally.
- And Flow again become zero after 3.25 to 5 second.



Area function of spool displacement

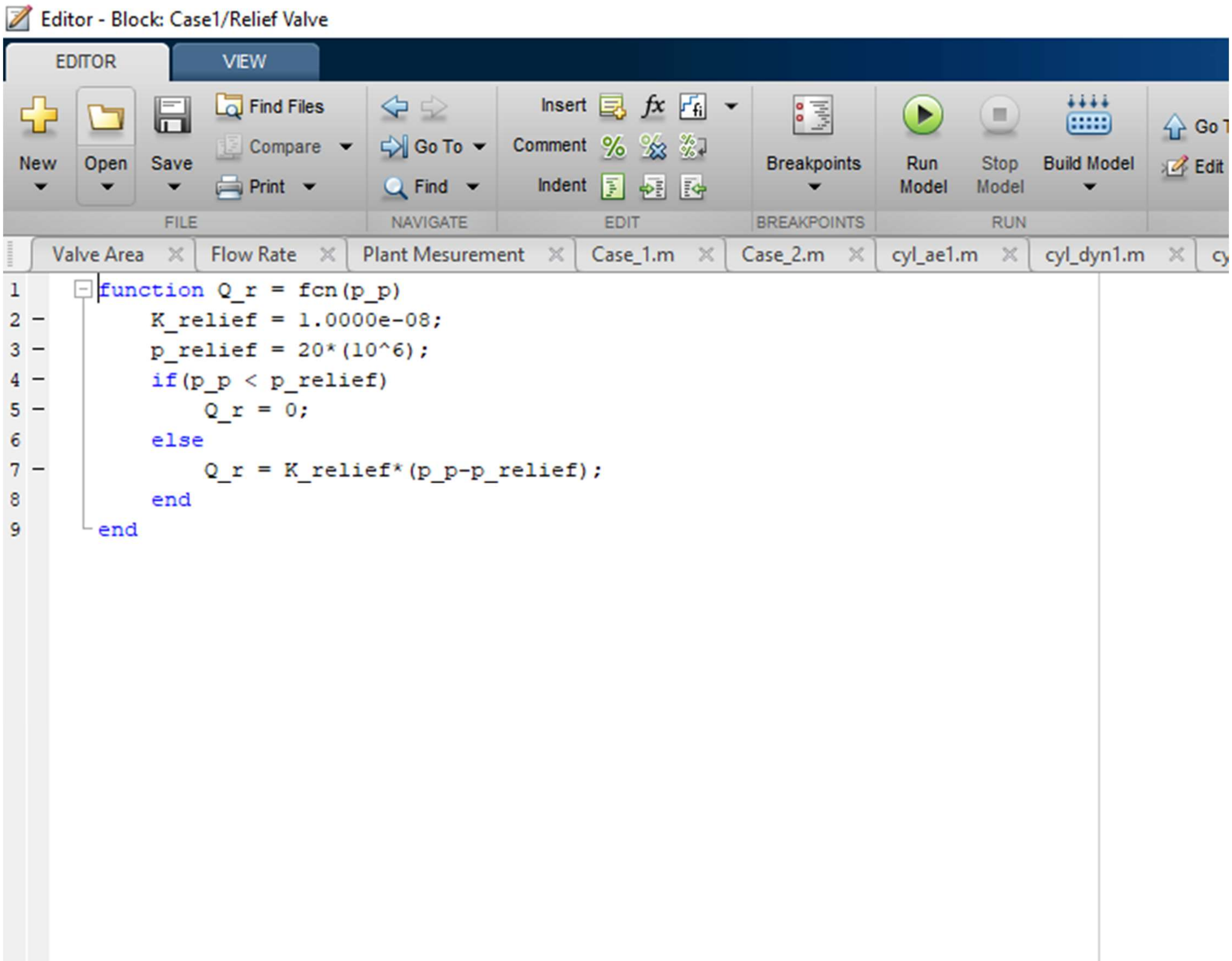
- This File shows Linear Relationship between Spool Displacement and Change in Area takes place as function of spool displacement.




```
function [Q_pa,Q_pb,Q_at,Q_bt,Q_p,Q_pt] = fcn(p_p, p_a, p_b, A_pa, A_pb, A_at, A_bt, p_t, x_s)
    C_d = 0.0650;
    D_p = 0.0001;
    w_pump = 25;
Q_p = D_p * w_pump ;
Q_pt = 0.0 ;
    if(x_s >= 0)
        Q_pa = C_d*A_pa*(abs(p_p-p_a))^0.5;
        Q_bt = C_d*A_bt*(abs(p_b-p_t))^0.5;
        Q_pb = 0;
        Q_at = 0;
    else
        Q_pb = C_d*A_pb*(abs(p_p-p_b))^0.5;
        Q_at = C_d*A_at*(abs(p_a-p_t))^0.5;
        Q_pa = 0;
        Q_bt = 0;

    end
end
```

Flow rates

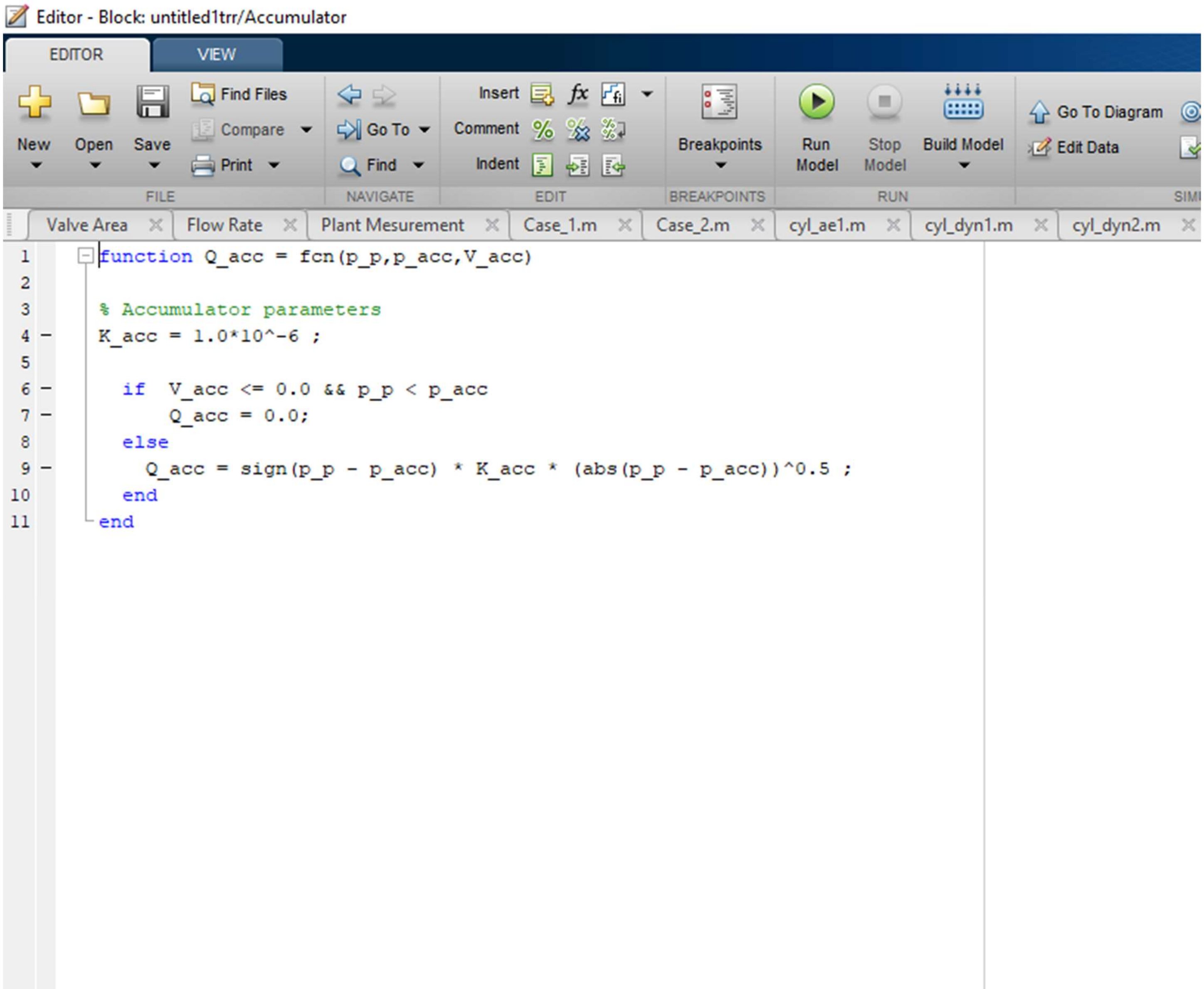
- This file shows the flowrate to cylinder as function of Area that is open depends upon spool position.
- One Part is for Up motion of cylinder and other part is for down motion of cylinder.
- As spool position is greater than 0,there will be flow from pump to A side of cylinder so cylinder will go down side which will cause the flow from B side of cylinder to Tank and From pump to B side of Cylinder , there is zero flow rate because all flow goes to A side and same for A to tank.




```
function Q_r = fcn(p_p)
    K_relief = 1.0000e-08;
    p_relief = 20*(10^6);
    if(p_p < p_relief)
        Q_r = 0;
    else
        Q_r = K_relief*(p_p-p_relief);
    end
end
```

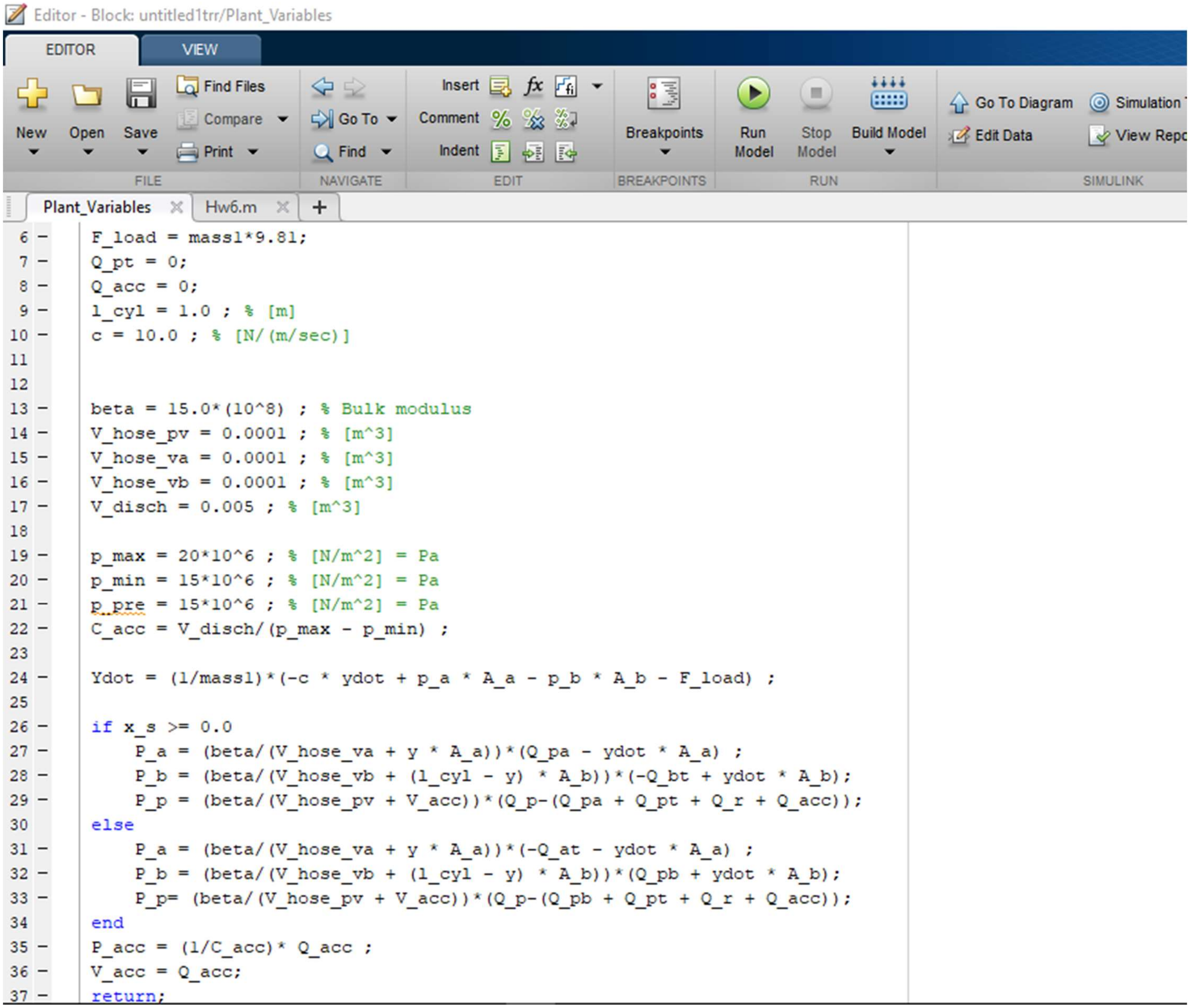
Relief Valve

- This File shows how relief valve works and in which conditions , relief valve opens to dump excessive flow
- Basically when pump pressure us increases greater than relief valve pressure in our example it is given, Relief valve opens which basically puppet type valve and puppet moves its position because of excessive pressure and it dumps excessive flow to the Tank or else there is no flow through relief valve in normal condition.



Accumulator

- This Files shows the flow rate through accumulator when needed
- Accumulator generally used for reducing pressure spikes and supplying flow when needed.
- It works same as Capacitor as it stores the some amount of and when needed it provides.

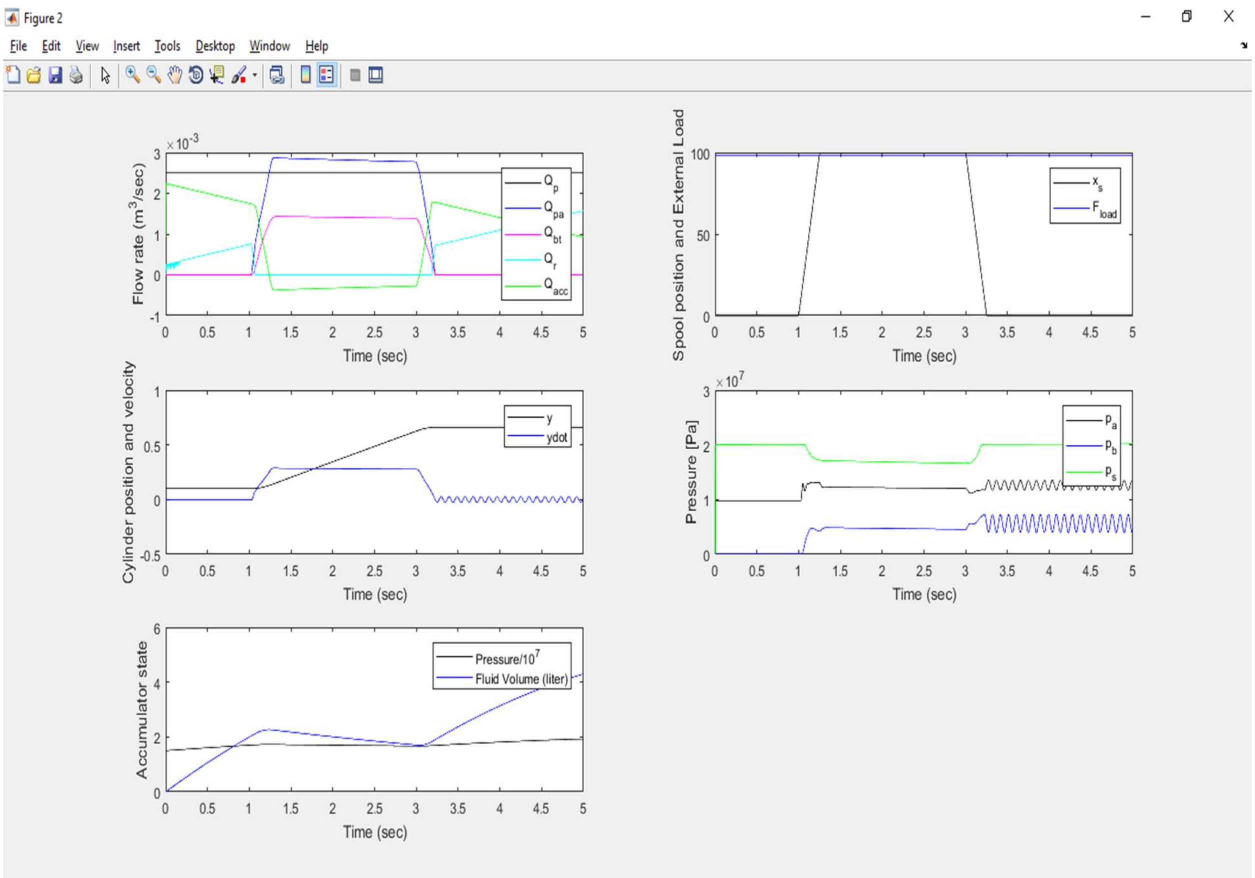


```
F_load = mass1*9.81;
Q_pt = 0;
Q_acc = 0;
l_cyl = 1.0 ; % [m]
c = 10.0 ; % [N/(m/sec)]


beta = 15.0*(10^8) ; % Bulk modulus
V_hose_pv = 0.0001 ; % [m^3]
V_hose_va = 0.0001 ; % [m^3]
V_hose_vb = 0.0001 ; % [m^3]
V_disch = 0.005 ; % [m^3]

p_max = 20*10^6 ; % [N/m^2] = Pa
p_min = 15*10^6 ; % [N/m^2] = Pa
p_pre = 15*10^6 ; % [N/m^2] = Pa
C_acc = V_disch/(p_max - p_min) ;

Ydot = (1/mass1)*(-c * ydot + p_a * A_a - p_b * A_b - F_load) ;

if x_s >= 0.0
    P_a = (beta/(V_hose_va + y * A_a))*(Q_pa - ydot * A_a) ;
    P_b = (beta/(V_hose_vb + (l_cyl - y) * A_b))*(-Q_bt + ydot * A_b);
    P_p = (beta/(V_hose_pv + V_acc))*(Q_p-(Q_pa + Q_pt + Q_r + Q_acc));
else
    P_a = (beta/(V_hose_va + y * A_a))*(-Q_at - ydot * A_a) ;
    P_b = (beta/(V_hose_vb + (l_cyl - y) * A_b))*(Q_pb + ydot * A_b);
    P_p= (beta/(V_hose_pv + V_acc))*(Q_p-(Q_pb + Q_pt + Q_r + Q_acc));
end
P_acc = (1/C_acc)* Q_acc ;
V_acc = Q_acc;
return;
```

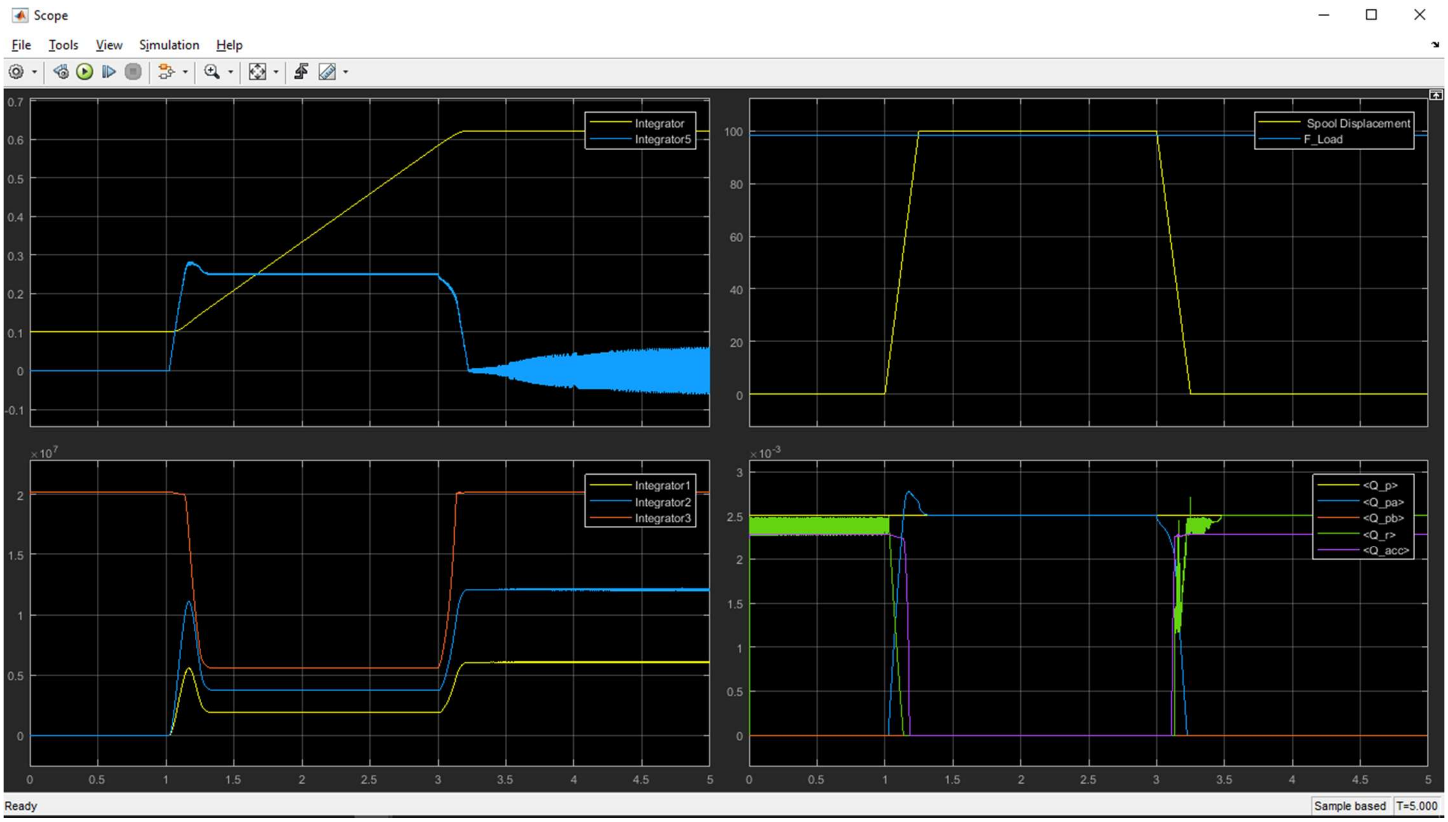
Plant Output

- This files shows how we can achieve desired output that we require.
- Interested parameters are Position and velocity of cylinder piston, Pressure at Side A, B and pump pressure, pressure from Accumulator and volume of Accumulator.
- As it is closed loop control, so this system has some kind of position measurement sensors and pressure measurement sensors in order to feed back again for calculation.

Here I have attached Matlab result for Case 2.



Simulink Result for Case 2



<u>Observation:</u>

Result from both matlab and Simulink file is similar and there is not much difference.

<u>Conclusion</u>

By Comparing Case 1 and Case 2, Following **Conclusions** are made.

- From Figures, In first charge for Flow-rate vs Time, we can see that as in case 1 there is no accumulator, so there is no control for flowrate and flowrate to cylinder is maximum up to rated flowrate only. On the other if you see case 2, we have included accumulator and it is adjusting flowrate when there is high need and also controls the pressure spikes.
- In second figure, we are getting some spikes in case 2 just because of inertia that we have included in case 2, where as in case 1, we have taken ideal case.
- Spool position with external load diagram is almost same in both case because there is not much effect of spool dynamic response that we took in Case 2 as first order filter and it is almost behaving like ideal case.
- Change in high frequency content of the pressure signals due to the accumulator. As expected, accumulator helps damp-out the pressure spikes.